INTERPLANETARY GUIDANCE SYSTEM REQUIREMENTS STUDY

VOLUME II

COMPUTER PROGRAM DESCRIPTIONS

PART 1

PROGRAM DESCRIPTIONS FOR

INTERPLANETARY FREE-FALL TRAJECTORIES

Prepared for
ERC Systems Laboratory
under Contract NAS 12-7

AC ELECTRONICS DIVISION
General Motors Corporation
El Segundo, California

TABLE OF CONTENTS

ABSTRACT

This document contains the description of the digital simulations for the generation of nominal interplanetary free-fall trajectories. It is organized in three parts. Part 1-1 and Part 1-2 contain the description of two computer decks for heliocentric and planetocentric phases, respectively. Part 1-3 gives an outline of how these programs can be used for the design of interplanetary free-fall mission profiles.

PART 1-1

PROGRAM DESCRIPTION

FOR

DETERMINATION OF HELIOCENTRIC

AND

PLANETOCENTRIC ORBITS

ABSTRACT

This section contains the description of a digital computer program for the design of interplanetary nominal free-fall trajectories based upon two-body sphere-of-influence techniques. In combination with Program 281, it provides the initial conditions of the nominal trajectory for the performance assessment simulation of aided-inertial space guidance systems during the free-fall phases of an interplanetary mission (Program Deck 284.0).

TABLE OF CONTENTS

TABLE OF CONTENTS

1.0 INTRODUCTION

The mathematical model for the performance assessment of space guidance systems during free-fall phases requires a nominal trajectory. Two conceptually different methods are frequently used to determine the nominal free-fall trajectory of an interplanetary mission. The first method uses the "exact" dynamical equations, i.e., the influence of all planets in the solar system upon the space vehicle are taken into account. In this case, no "closed form" analytical solutions do exist and the trajectory is determined by numerical integration of the associated set of nonlinear differential equations. The second method is an approximate one. It utilizes the fact that the vehicle's motion in different portions of free-fall is essentially determined by the gravitational attraction of a single body. This means that the motion can be described by different two-body orbits in different portions of the flight; heliocentric ellipses in the transfer phase, planetocentric ellipses or hyperbolas during planetary approach, planet departure phases, and periodic orbits around a planet. If the different two-body orbits are appropriately "matched", the resulting orbit constitutes, as experience has shown, a sufficiently accurate approximation to the exact orbit. [1]*

For the present study, it is particularly important that the approximation in the nominal trajectory does not influence the conclusions about the adequacy or inadequacy of a selected guidance system configuration for the particular mission. Thus, the deviations of the approximate trajectory from the exact trajectory must be minimized in those parameters which have the strongest influence upon the guidance system performance. The two most important parameters are the time of flight and the distance from navigational references, such as planets, especially during those phases in which the most useful information is obtained. For these reasons, the time of flight between the closest approach at two consecutive planets and the closest approach distance itself is the same for the exact and the approximating orbit. Simulation results have shown that such an approximation procedure does not influence the conclusions about the adequacy of the performance of a selected system configuration. Reduction in computer time and elimination of the numerical errors inherent in any integration process are additional advantages of this approximation.

The program itself is divided into two sections, namely, the heliocentric and planetocentric computations. The heliocentric portion utilizes the JPL ephemeris routine [2] in conjunction with Lambert's theorem. Any number of planets, departure and arrival dates, can be specified. The orbital parameters of the corresponding heliocentric transfer ellipses and the associated hyperbolic excess velocity vectors are computed. Lambert's theorem is implemented in such a fashion that conjunction as well as opposition class trajectories can be computed. Only the case in which the two heliocentric position vectors are collinear within a certain ϵ-bound is excluded since the orbit plane is not specified in this case. In the planetocentric block, the hyperbolic approach and departure trajectories are computed. Three different cases have to be distinguished, namely, the regular case, planetary departure, and planetary entry.

* Numbers in brackets indicate references at the end of the section.

The regular case is characterized by the fact that two hyperbolic excess velocity vectors are available and the planetocentric orbit-plane is, therefore, determined. The incoming hyperbola is determined by the incoming hyperbolic excess velocity vector and a prespecified closest approach distance. The outgoing hyperbola is then computed using the outgoing hyperbolic excess velocity vector and the position vector which constitutes pericenter for the incoming hyperbola. In general, there will be a discontinuity in the velocity at the intersection of these two hyperbolas which has to be provided either by the propulsion system or atmospheric braking maneuvers. It should be emphasized that this arbitrary selection of the point on the incoming hyperbola for the trajectory change maneuver does not necessarily constitute an optimum. Using this option of the program implies that the parking orbit in planetary stopover missions is coplanar with the approach and departure hyperbola. However, out-of-plane parking orbits can easily be treated by using the planetary departure option of the program which is explained below.

For planetary departure phases, only one hyperbolic excess vector is provided by the heliocentric computation. In order to define the planetocentric orbit plane, a second vector must be specified. Two options are available. In the first option, the projection of the hyperbolic excess velocity vector on the equatorial plane is rotated clockwise by $\pi/2$. In the second option, this vector can be chosen in an arbitrary manner.

Analogous to the planetary departure phase, only one hyperbolic excess velocity vector is given for planetary entry phases. The plane of motion is specified by additional constraints; namely, flightpath angle, the declination of the pericenter at which the closest approach occurs, and the constraint implying direct motion. In all these cases, the orbital parameters of the hyperbolas are computed. This essentially describes the capabilities of the Program 291.1.

The actual initial conditions for the matched conics are determined by using the output of this program in conjunction with Program 281. This is done by the following procedure: first, the position and velocity for each planetocentric hyperbola is computed at the point of closest approach and the sphere of influence, thus providing the initial conditions for the planetocentric flight phases. The associated flight times are also determined. Corresponding heliocentric transfer ellipses which are matched in position at the spheres of influence are found using the output of 281 in 291.1 in two steps. First, modified ephemerides of the planets are computed taking into account the flight times in the planetocentric phase. Then the new heliocentric position vectors of the probe at corresponding points of the spheres of influence and the associated times are used to find a new heliocentric transfer ellipse. This yields the initial conditions for the heliocentric phases and completes the process. The procedure could be repeated by using the new hyperbolic excess velocity vectors in conjuction with 281 in order to modify the planetocentric hyperbolas and achieve matching in position and velocity. However, for the present purposes, the "first order" matching is sufficiently accurate.

2.0 MATHEMATICAL MODEL

2.1 A GENERAL DESCRIPTION OF THE PROBLEM

The design of free-fall trajectories for interplanetary missions requires knowledge of the position and velocity of the planets as function of time. The information is provided in this program by the new JPL ephemeris routine (EPHEM) [2].

Specification of Julian dates and corresponding code names for the planets in the program will provide the desired information.

Then, given two planets and two dates, Lambert's equation may be used to compute the heliocentric elliptic trajectory required to connect the two planetary positions in the time interval defined by the difference in the dates supplied. As planetary velocities are known, the $\underline{V}_h$ (hyperbolic-excess velocity) vectors are readily obtained by differencing the planetary and trajectory velocities.

In the case of a round-trip or multi-planet mission, there will exist at each target planet two $\underline{V}_h$ vectors—one of which corresponds to the incoming heliocentric transfer ellipse and the other of which corresponds to the outgoing transfer ellipse. From very simple geometric considerations of central-force field motion, it immediately becomes clear that these two velocity vectors, originating at their respective planetocentric positions, will define a unique orbit plane which passes through the center of the target planet. Also, from the magnitude of $\underline{V}_h$ and the gravitational constant of the planet, one may obtain the semi-major axis of the planetocentric hyperbola. Finally, the hyperbola is specified completely by choosing a pericenter distance; time may be referenced to the arrival date.

This, in general, describes the basic approach to trajectory computations as employed in the design of Program 291.1.

2.2 FORMULATION FOR HELIOCENTRIC PHASE

A drawing of the practical basis for the heliocentric trajectory formulation scheme is illustrated in Figure 1.

In this, a typical case, a vehicle leaves planet 1 at time T_1 along the ellipse shown. It then arrives at planet 2 at time T_2 with the incoming and outgoing $\underline{V}_h$ vectors indicated. At planet 2 a velocity change is executed (this is not always necessary) and the vehicle leaves planet 2 on a second ellipse and arrives at planet 1 at time T_3. If $T_3 - T_1$ is equal to an integral multiple of the orbital period of planet 1, the resulting transfer ellipses will, in fact, be identical and have continuous first derivatives with respect to time over the interval $T_1 \leq t \leq T_3$. Such a trajectory is called a mono-elliptic transfer. In general, however, it will be necessary to "break up" each round-trip mission into two or more distinct transfer ellipses, each of which is called a "leg". In this analysis, it is assumed that the only source of gravitational fields is that of the sun, and the masses of the planets can be neglected during the helio-centric transfer phase.

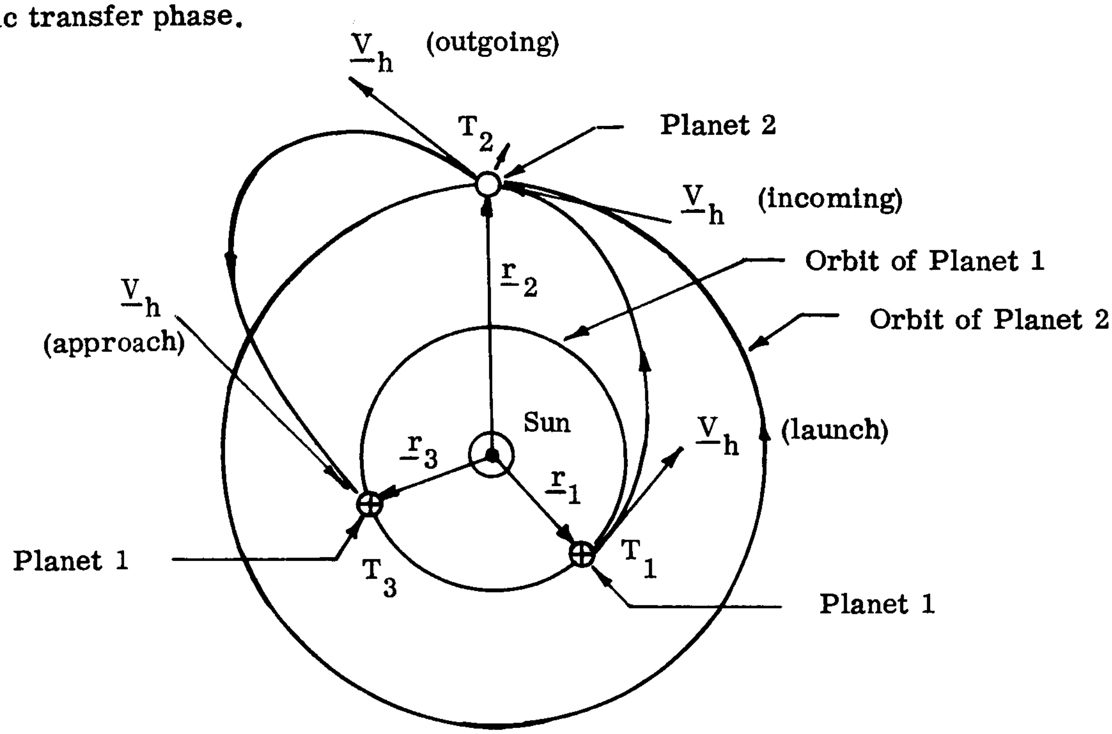


Figure 1. Geometry of a Typical Interplanetary Round-trip Mission

The heliocentric transfer phase of this program is formulated in accordance with the philosophy described above, i.e., each leg of a mission is computed separately. Thus, it is necessary at all times to enter at least <u>two</u> planets and two dates for proper computation.

2.2.1 Ephemeris Routine, EPHEM

This consists of a tabulation of ephemeris data for all nine planets and the Moon from December 30, 1949 (J.D. 2433280.5) to January 5, 2000 (J.D. 2451548.5). The position and velocity of any planet at any time within the interval specified is found from Everett's interpolation formula utilization fourth differences [2].

Thus, it is apparent that specification of T_1, T_2, planet 1, and planet 2 is all that is necessary to obtain $\underline{r}_1$, $\underline{r}_2$ and the respective planetary velocities (Figure 1). All vectors are expressed relative to the mean equator and equinox of 1950.

2.2.2 Lambert's Equation [3]

Lambert's equation is used in the form required to compute an ellipse given two position vectors ($\underline{r}_1, \underline{r}_2$) and the flight time between them ($T_2 - T_1$). The basic equation assumes one of four forms shown below, depending upon the conditions of the problem:

1. $$T_c = \frac{P}{2\pi}\left[(\alpha - \sin\alpha) - (\beta - \sin\beta)\right] + NP$$

2. $$T_c = \frac{P}{2\pi}\left[(\alpha - \sin\alpha) + (\beta - \sin\beta)\right] + NP$$

3. $$T_c = P - \frac{P}{2\pi}\left[(\alpha - \sin\alpha) + (\beta - \sin\beta)\right] + NP$$

α and β are computed from the following:

$$\cos\alpha = 1 - \frac{s}{a}$$

$$\cos\beta = 1 - \frac{s - c}{a}$$

where

$$s = \frac{r_1 + r_2 + c}{2},$$

$$c = \sqrt{r_1^2 + r_2^2 - 2r_1r_2\cos\theta},$$

and

$\theta = \sphericalangle\ \underline{r}_1, \underline{r}_2$, commonly called the <u>transfer angle.</u>

P is the period of the ellipse given by

$$P = 2\pi\sqrt{\frac{a^3}{\mu}}$$

In this case, a is obtained by iteration from a predetermined first guess. The program logic determines one of the four forms of the equation to use based upon the specified flight time T and the criterion of direct motion. N is the number of complete circuits encompassed by the probe. In cases where $N > 0$, several solutions to the equations are possible, and it is necessary for the user to know the approximate value of a for the desired trajectory. In practice, the input value is usually made about 1 percent higher than the desired value, as the iteration scheme tends to converge toward the lower value of an initial a. The detailed equations and logic flow are specified in Section 5.1.2.

2.2.3 Conic Determination

The plane of motion is determined from $\underline{r}_1$ and $\underline{r}_2$ such that W_z, the polar component of $\underline{W}$, is positive. The semilatus rectum is determined from the equation

$$p = [4a\,(s - r_1)(s - r_2)/c^2]\ \sin^2\,[(\alpha \pm \beta)/2]$$

where the sign in $\alpha \pm \beta$ is chosen by program logic (see Section 5.1.1).

The true anomolies V_1 and V_2, corresponding to $\underline{r}_1$ and $\underline{r}_2$ respectively, are obtained from the simultaneous equations

$$e \cos V_1 = \frac{p}{r_1} - 1$$

$$e \cos V_2 = \frac{p}{r_2} - 1$$

Since $V_2 - V_1$ is the transfer angle, there results two equations in two unknowns which may be solved explicitly for p and $V_2 - V_1 = \Delta V$.

The elliptic velocity $\underline{V}$ is obtained from the following sequence of computations:

$$\dot{r} = \sqrt{\frac{\mu}{p}}\ e \sin V$$

$$\underline{U}_v = \underline{W} \times \frac{\underline{r}}{r}$$

$$\underline{V} = \dot{r}\,\frac{\underline{r}}{r} + \frac{\sqrt{p\mu}}{r}\ \underline{U}_v$$

In this case r may be r_1 or r_2 in accordance with the situation. These equations resolve the velocity into two components, one radial and the other normal to the radius vector $\underline{r}$.

The $\underline{V}_h$ or hyperbolic-excess velocity vectors are obtained by differencing the elliptic and planetary velocities and transforming the results to spherical coordinates for the output. This is accomplished through the equations

$$\underline{V}_h = \underline{V} - \dot{\underline{r}}_p$$

$$\phi = \sin^{-1} V_{hz}$$

$$\theta = \tan^{-1} \frac{V_{hy}}{V_{hx}}$$

where

$\underline{V}_h$ = hyperbolic-excess velocity

$\underline{V}$ = velocity of vehicle on central body ellipse

$\dot{\underline{r}}_p$ = velocity of planet relative to central body

ϕ = declination of hyperbolic-excess velocity vector

θ = right ascension of hyperbolic velocity vector.

2.3 PLANETOCENTRIC FORMULATIONS

As stated earlier, the basic problem in the planetocentric section is that of computing a hyperbolic trajectory given the hyperbolic-excess velocity vectors and closest-approach distance.

In the case of round-trip interplanetary trajectories, there will usually result two hyperbolas at the target planet; these must be matched in position and time. One hyperbola (incoming) is defined by the incoming $\underline{V}_h$ vector ($\underline{V}_{hp}$), pericenter distance, and direction of the outgoing asymptote. The other hyperbola (outgoing) is defined by the outgoing $\underline{V}_h$ ($\underline{V}_{hL}$), common orbit plane, and closest-approach position on the incoming hyperbola. Although these formulations assume that $\underline{V}_h$ is at infinity, as the radii of planetary spheres of influence are quite large, the resulting discrepancy will be small for purposes of mission analysis.

Normally the planetocentric computations are performed only for the first planet in a leg. However, if the second planet is EARTH, computations are made for both planets, using a special set of equations for Earth re-entry. Also, if the first planet is EARTH, the orbit plane is computed in a unique manner. These special cases are explained in more detail in the following sections.

In addition to the fundamental computations mentioned above, the program will also compute additional parameters which may be useful as information for design purposes. These include such things as the asymptote deflection angle K, optimum closest-approach distance r_{opt}, angle between the incoming and outgoing hyperbolic pericenters ω, and polynomial coefficients for the computation of the optimum radius for circular capture orbits (a_1, a_2, a_3, a_4).

2.3.1 Orbit Plane Determination

2.3.1.1 Orbit Plane for Planetary Departure (See Figure 2)

Let $\underline{S}_1$, $\underline{S}_2$ be such that $\underline{S}_1 \times \underline{S}_2 / |\underline{S}_1 \times \underline{S}_2| = \underline{W}$. Further, let $\underline{S}_2 = \underline{V}_{hL} / |\underline{V}_{hL}|$.
If the first planet is EARTH, then the $\underline{S}_1$ vector is chosen in such a way that

1. The resulting motion is direct rather than retrograde,
2. The orbit plane inclination is a minimum for a given $\underline{V}_{hL}$

These conditions will insure easterly launchings and, in addition, they will make maximum use of the earth's rotational speed as an aid in minimizing the required boost velocity. This follows from the fact that maximum tangential speed due to earth rotation is obtained whenever the orbit plane is close to the equatorial plane. In turn, it can be shown that two sufficient conditions for W_z to be a maximum (and hence the orbit plane inclination a minimum) for a given $\underline{S}_2$ are

1. That $\phi_1 = 0$, where ϕ_1 is the declination of $\underline{S}_1$, and
2. $\tan \theta_1 = - \text{ctn}\, \theta_2$, where θ_1 and θ_2 are respectively the right ascensions of $\underline{S}_1$ and $\underline{S}_2$.

All of the criteria above may be satisfied by choosing $\underline{S}_1$ such that its declination is zero, and its right ascension is 90° less than the right ascension of $\underline{S}_2$. Additionally, $\underline{S}_1$ chosen under these conditions is in effect the line of the ascending node. FLAG 3 enables the user to input arbitrary values of $\underline{S}_1$ if a different criterion is desired. This flag will work for any planet.

2.3.1.2 Orbit Plane at Target Planet

If, for a round-trip flyby trajectory, the orbit plane at the target planet is chosen such that it contains the incoming and outgoing $\underline{V}_h$ vectors, no plane change will be necessary to complete the mission. As this criteria is undoubtedly the most expedient situation with regard to trajectory change maneuvers, it has been incorporated into this program*. The planetocentric orbit plane vector $\underline{W}$ is then computed from

$$\underline{W} = \frac{\underline{S}_1 \times \underline{S}_2}{|\underline{S}_1 \times \underline{S}_2|}$$

where

$$\underline{S}_1 = \frac{\underline{V}_{hp}}{|\underline{V}_{hp}|} \quad ; \quad \underline{S}_2 = \frac{\underline{V}_{hL}}{|\underline{V}_{hL}|}$$

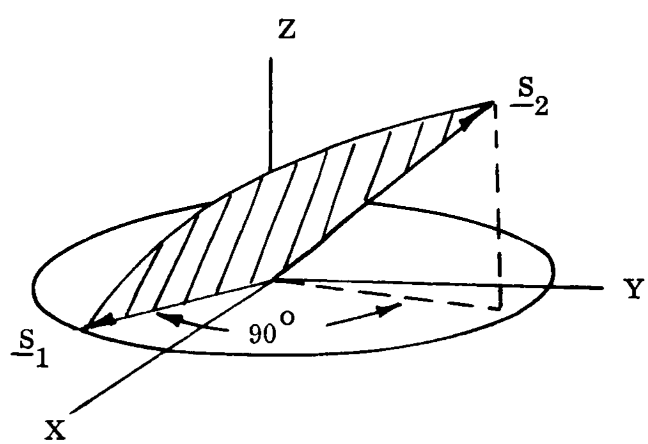


Figure 2. The Orbit Plane of Minimum Inclination

* Nevertheless, an option (Flag 3) has been provided to allow selection of a different orbit plane if the user so desires.

2.3.1.3 Earth Re-entry

The determination of the orbit plane for Earth re-entry is much more complicated than those explained in the previous two cases. Basically, the objective is that of orienting the vehicle velocity at atmospheric contact such that the vehicle will tend to travel under gravity toward a specified declination relative to the Earth. This problem is illustrated in Figures 3 and 4.

Let V_r be the speed at a re-entry distance of r_r; let γ be the angle between $\underline{V}_r$ and $\underline{r}_r$. Then

$$V_r = \sqrt{\mu \left(\frac{2}{r_r} - \frac{1}{a}\right)}$$

where

$$\frac{1}{a} = - \frac{V_h^2}{GME}$$

V_h being the hyperbolic excess speed, a the semi-major axis of the hyperbola, and GME the gravitational constant of the Earth. As it is usually required that γ and r_r be specified in order to obtain a desired re-entry corridor, the eccentricity e may be computed from

$$e = \sqrt{1 + \frac{r_r \sin^2 \gamma}{|a|^2} (2 |a| + r_r)}$$

If $\underline{S}$ is a unit vector in the direction of the incoming asymptote, $\underline{P}$ is the perifocus vector (see Figure 3), and α is the angle between them, one may formulate the relationship

$$\cos \alpha = \underline{P} \cdot \underline{S} = S_x \cos \phi \cos \theta + S_y \cos \phi \sin \theta + S_z \sin \phi = \frac{1}{e}$$

where S_x, S_y, S_z are the Cartesian components of $\underline{S}$, ϕ is a specified declination equal to the declination of $\underline{P}$, and θ is the right ascension of $\underline{P}$. Clearly, in the latter equation, θ is the only unknown and the equation may be solved, yielding the solutions

$$\sin\theta = \frac{C S_y \pm S_x \sqrt{S_x^2 + S_y^2 - C^2}}{S_x^2 + S_y^2}$$

$$\cos\theta = \frac{C S_x \pm S_y \sqrt{S_x^2 + S_y^2 - C^2}}{S_x^2 + S_y^2}$$

where

$$C = \frac{1}{\cos\phi}\left(\frac{1}{e} - S_z \sin\phi\right)$$

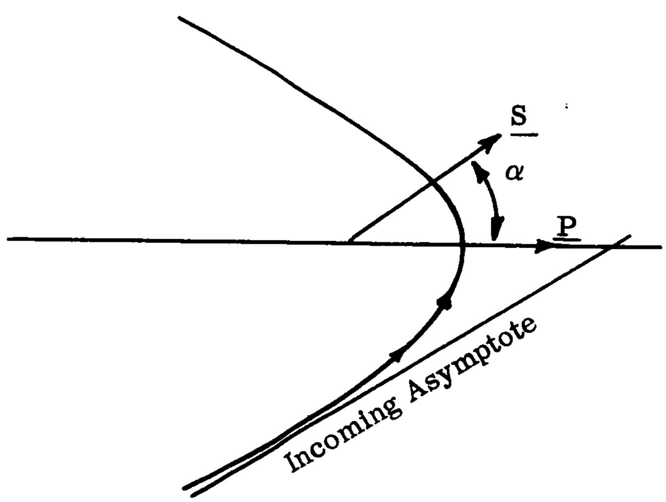


Figure 3. Location of S and P Vectors

Obviously solutions exist in two quadrants, and the one corresponding to $W_z \geq 0$ is chosen. Thus θ and ϕ complete the definition of P, and it, together with S, defines the orbit plane. Notice that an Earth-fixed longitude is clearly a function of time, and for this reason it was not considered in formulating the problem. For certain choices of ϕ it is possible for the radicand to become negative. If such should be the case, the program will print the comment "RE-ENTRY CONDITIONS CANNOT BE OBTAINED".

By appropriate handling of input parameters, these computations can also be extended to re-entry at other planets.

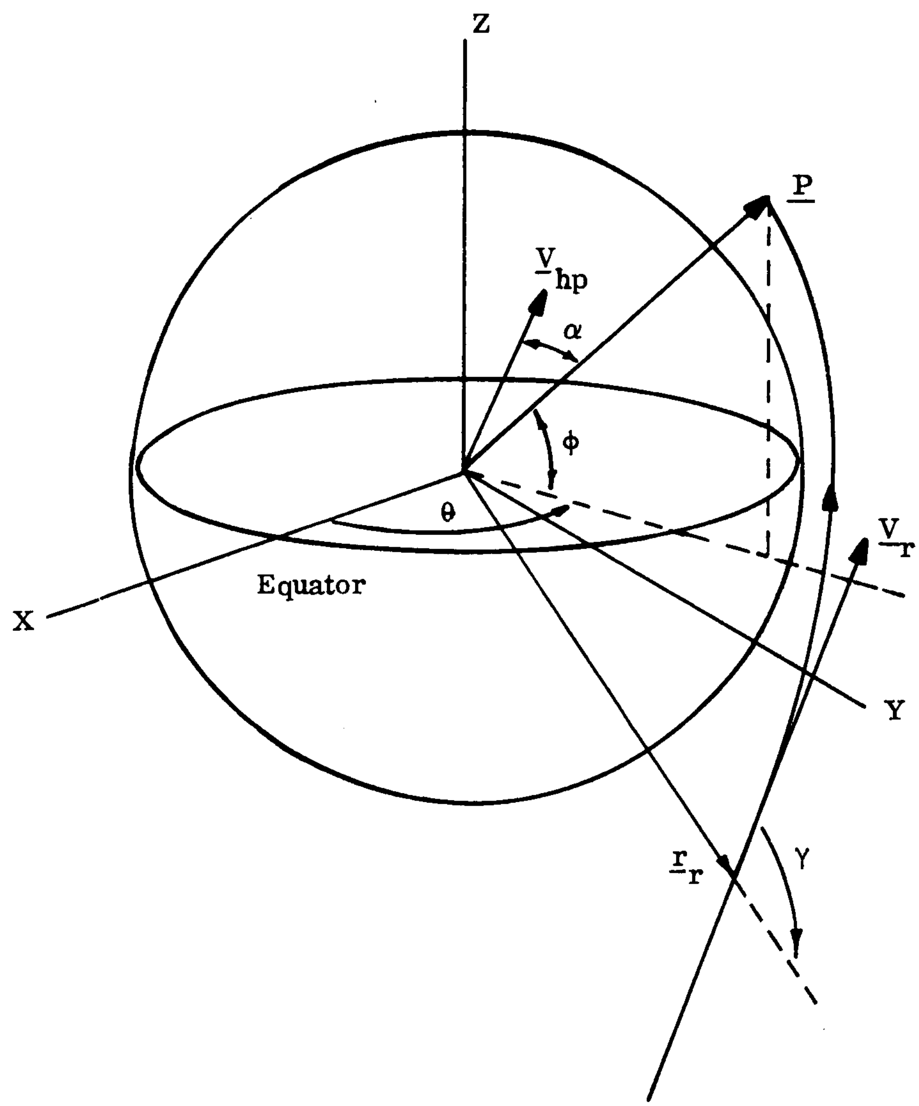


Figure 4. Geometry of Hyperbola at Re-entry

2.3.2 Trajectory Parameters at the Target Planet

Figure 5 illustrates the geometry of the trajectory at the target planet. This will usually consist of two distinct hyperbolas which are matched in position and time at $\underline{r}_p$, the closest-approach vector for the first hyperbola. K is the angle between the unit asymptote vectors $\underline{S}_1$, $\underline{S}_2$, while $\underline{r}_p'$ is the closest-approach vector for the outgoing hyperbola. ω is the angle between $\underline{r}_p$ and $\underline{r}_p'$.

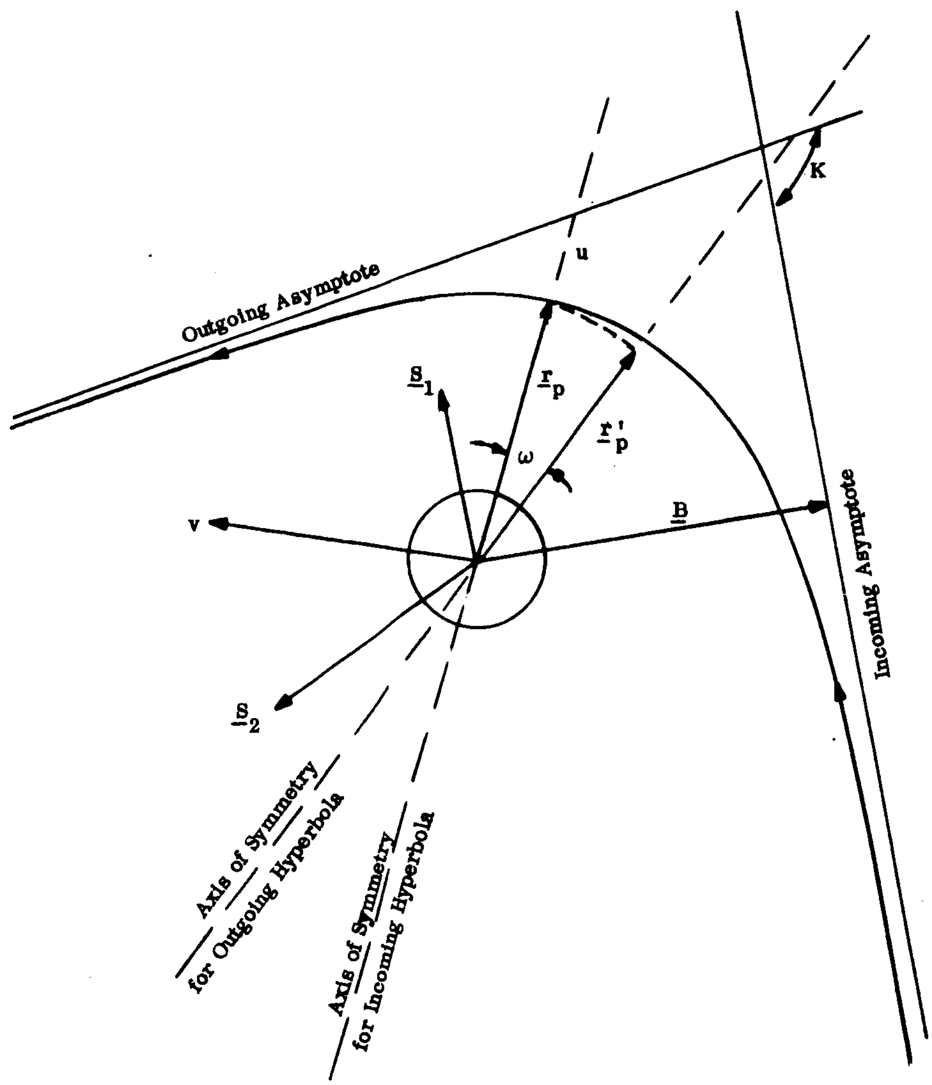


Figure 5. Incoming and Outgoing Trajectories

2.3.2.1 The Incoming Trajectory

As $\underline{W}$, $\underline{S}_1$, and $\underline{S}_2$ have already been defined in Section 2.3.1.2, the remaining parameters may be computed for the incoming hyperbola from these and V_{hp}, the incoming hyperbolic-excess speed. These include a, the semi-major axis, computed from

$$a = - \frac{GM}{V_{hp}^2}$$

where GM is the gravitational constant of the planet; the eccentricity e from

$$e = 1 - \frac{r_p}{a}$$

and B, the "impact parameter", as expressed by

$$B = \sqrt{\frac{2\ GM\ r_p}{V_{hp}^2} + r_p^2}$$

The vectors $\underline{R}$, $\underline{S}_1$, and $\underline{T}$ form a special orthogonal set such that

$$\underline{R} = \underline{S}_1 \times \underline{T}$$

T lies in the XY plane and is defined by

$$\underline{T} = \frac{S_{1y}}{\sqrt{S_{1x}^2 + S_{1y}^2}}, \quad \frac{-S_{1x}}{\sqrt{S_{1x}^2 + S_{1y}^2}}, \quad 0$$

The quantities $\underline{B} \cdot \underline{T}$ and $\underline{B} \cdot \underline{R}$ are sometimes useful in trajectory computation and design.

The perifocus unit vector $\underline{P}$ is obtained from the linear combination of $\underline{S}_1$ and $\underline{S}_3$ as

$$P = \sqrt{\frac{e^2 - 1}{e^2}}\ \underline{S}_3 + \frac{1}{e}\ \underline{S}_1$$

where $\underline{S}_3 = \underline{S}_1 \times \underline{W}$

2.3.2.2 The Outgoing Trajectory

The problem of determining the outgoing trajectory is that of defining a hyperbola given one position $\underline{r}_p$ and the hyperbolic-excess velocity vector. This may be determined from the polar equation of a conic

$$e\, r \cos f + a\, e^2 + r - a = 0$$

where f is the true anomaly; and the equation

$$\cos \alpha = \frac{1}{e'} = \frac{\underline{S}_2 \cdot \underline{r}'_p}{|\underline{r}'_p|}$$

where α is the angle between $\underline{r}'_p$ and $\underline{S}_2$, and e' is the eccentricity of the hyperbola to be computed. The quantities S_u and S_v are formed, where

$$S_u = -\underline{S}_2 \cdot \underline{P}$$

$$S_v = -\underline{S}_2 \cdot (\underline{P} \times \underline{W})$$

Then, from Figure 5 it is apparent that relative to the u, v coordinate system shown, $\underline{r}'_p/|\underline{r}'_p|$ has the respective components $\cos \omega$ and $\sin \omega$. Hence one may establish the relationship

$$\frac{1}{e'} = \left(\frac{\underline{S}_2 \cdot \underline{r}'_p}{|\underline{r}'_p|} \right)_{u,v} = S_u \cos \omega + S_v \sin \omega$$

Since, when $r = r_p$, $f = \omega$, $\cos f = \cos \omega$, and the conic equation may be written as

$$e'\, r_p \cos \omega + a'\, e'^2 + r_p = 0$$

where a' is the semi-major axis of the hyperbola in question and is computed from the outgoing V_h.

From the equation

$$S_u \cos \omega + S_v \sin \omega = \frac{1}{e'}$$

one may solve for $e' \cos \omega$ and substitute this into the conic equation, yielding

$$r_p (S_u + 1) + r_p S_v \sqrt{e'^2 - 1} + a' (e'^2 - 1) = 0$$

This is readily solved using the transformation

$$z^2 = e'^2 - 1$$

which results in the quadratic

$$a' z^2 + r_p S_v z + r_p (S_u + 1) = 0$$

This in turn is solved for z^2 as

$$z^2 = \frac{r_p^2 S_v^2 + r_p S_v \sqrt{r_p^2 S_v^2 - 4a' r_p (S_u + 1)} - 2a' r_p (S_u + 1)}{2a'^2}$$

from which one may obtain e' and r_p' as

$$e' = \sqrt{z^2 + 1}$$

and

$$r_p' = a' (1 - e')$$

This completes all that is necessary to define the hyperbola.

2.3.2.3 Gravitational Turns

It is possible in many situations to use the gravitational attraction of the planet to deflect the vehicle such that its outgoing path will be directed parallel to the outgoing $\underline{V}_h$. This is accomplished through choice of the closest-approach distance $r_p = r_{opt}$. The relationship is

$$r_{opt} = \frac{GM}{|\underline{V}_{hp}|^2} \cdot \left(\csc\frac{K}{2} - 1\right)$$

where $\underline{V}_{hp}$ is the incoming hyperbolic-excess velocity and K is obtained from

$$K = \cos^{-1} \underline{S}_1 \cdot \underline{S}_2$$

In cases where r_{opt} is less than some minimum allowable value, the smallest practical distance for r_p should be used.

2.3.2.4 Use of ω

In some cases it may occur that $\underline{r}'_p$ lies ahead of $\underline{r}_p$ in the direction of travel, and further, that $|\underline{r}'_p| < |\underline{r}_p|$. This means that on the outgoing trajectory, the vehicle may come closer to the planet than originally planned, and the effect may or may not be serious. If such is the case, and if $|\underline{r}_p| - |\underline{r}'_p|$ is more than about one percent of r_p, the trajectory should be redesigned or a larger value of r_p should be used. In such cases it will be found that the angle ω will be quite large. In fact, ω may be used as a rough indication of the degree to which the incoming and outgoing hyperbolas are matched. In cases of perfect matching (no need for a velocity change) ω will be zero, and as the matching characteristics deteriorate, ω will increase.

2.3.2.5 Circular Capture Orbits

It can be shown that for two-impulse transfer from a hyperbolic to a circular to a hyperbolic orbit, the distance that yields the minimum total impulsive speed may be obtained from the real root of the quartic

$$16\, x^4 + a_1\, x^3 + a_2\, x^2 + a^3\, x + a_4 = 0$$

where

$$a_1 = 8(c_1 + c_2)$$

$$a_2 = -(c_1 - c_2)^2$$

$$a_3 = -2(c_1^2 c_2 + c_1 c_2^2)$$

$$a_4 = -c_1^2 c_2^2$$

$$c_1 = \frac{V_{hp}^2}{GM} \quad ; \quad c_2 = \frac{V_{hL}^2}{GM}$$

V_{hp} and V_{hL} are respectively the hyperbolic excess speeds of the incoming and outgoing hyperbolas. The optimum transfer distance r is obtained from x by

$$r = \frac{1}{x}$$

Program 291.1 computes the coefficients a_i but does not solve the quartic.

3.0 PROGRAM ORGANIZATION

Flow charts provide the basic framework around which the discussion is constructed. These diagrams serve to indicate the logical flow connecting different functional blocks. They do not describe literally the operation within the computer program itself because many of the programming details are of little interest to most engineers.

The flow charts have been arranged and drawn according to a heirarchical structure. The "highest" level, designated as Level I, depicts the over-all structure of the program. Each block appearing in this chart is described by another flow chart. These charts are designated as Level II. This policy is repeated for each block in every level until no further logic remains to be described.

Paragraph 3.1 contains a further discussion and definition of the criteria used to establish the different flow chart levels. The symbols used in the flow charts are defined in Paragraph 3.2. The symbols and nomenclature that are fundamental to the discussion and equations are defined in Paragraph 3.3.

3.1 SCHEMA FOR FLOW CHART PRESENTATION

As has already been stated, the flow charts are arranged according to "levels". In the resulting hierarchy, the Level I flow chart provides the most general description since it depicts the over-all program. Each functional block is further described by lower level flow charts. These charts indicate the logical flow within the block and describe the input and output requirements of the block. The equations used to obtain the desired outputs are presented as a supplement to the lowest level flow chart.

LEVEL I: This flow chart is designed to provide a very general description of the entire program. The titles assigned to the functional blocks are intended to be suggestive of the nature of the role to be performed within the block. Those functions that are to be performed in the basic computational cycle are designated by Roman numerals. Arabic symbols are used for functions that occur only once or play a passive role.

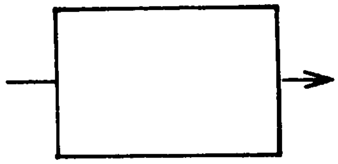
To indicate the basic logical decisions that can regulate and alter the flow between functional blocks, decision blocks are indicated. These blocks represent in a general manner the types of decisions that are required. The actual decision logic is described in the Level II flow charts of the functional blocks immediately preceding the decision block.

LEVEL II: The Level II flow charts provide the first concrete description of the program. Only the most important logical flow within each functional block is indicated on these diagrams. The quantities that are required for all logical and computational operations within this block are stated on this chart. These quantities are differentiated as being either INPUT (i.e., values provided initially by the engineer) or COMPUTED (i.e., values determined in other portions of the program). The quantities that are required in other parts of the program, either for print-out or for computations, are also indicated on this flow chart. The functional blocks that appear on these diagrams are denoted by two symbols (e.g., II.1 when discussing the "first" block in the Level II flow chart of functional block II) and a name. The names have been selected to provide some insight into the nature of the block.

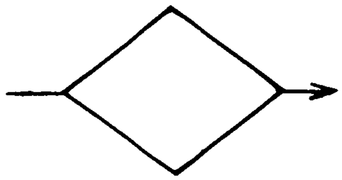
LEVEL III: These diagrams provide additional details of the logic flow within the functional blocks depicted at Level II. In this program definition, Level III provide the description of the most intimate logical details in almost every case so no purpose was served by proceeding to lower levels. These flow diagrams are augmented by the equations programmed into the computer. The input and output requirements of these blocks are stated on the diagrams. All of these quantities are summarized on the Level II flow chart.

3.2 DEFINITION OF FLOW CHART SYMBOLS

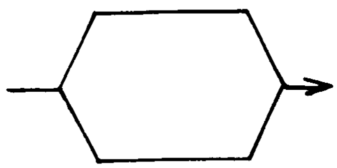
The following symbols represent the only ones that are used in the flow charts presented below.



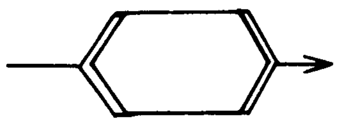
Set of operations that is to be described further by additional flow charts or by equations



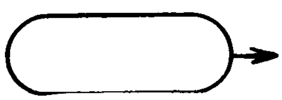
Logical Decision



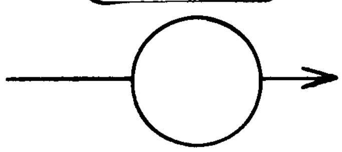
Operations that are predefined (i.e., in some other document)



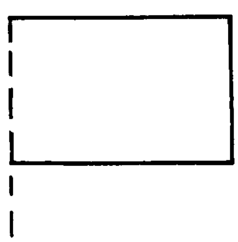
Operations are completely defined by the statements contained within the box



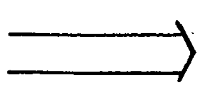
Connector used on Level II Flow Charts to indicate entry source and exit destination



Connector used on Level III flow charts



Summary of all quantities required in computations of flow chart on which this symbol appears or, alternatively, summary of all quantities computed in this flow chart which are required in other operations



This broad arrow appears on Level I and Level II flow charts. It is used to indicate information flow from one block to another. The more important information is stated within the arrow. This symbol has been introduced to emphasize that many quantities are transmitted between the functional blocks in the higher level charts.

3.3 DEFINITION OF MATHEMATICAL SYMBOLS

The following symbols are used in the description of this program:

3.3.1 Symbols for Heliocentric Equations

a	semi-major axis of ellipse in km
a_1	initial value for a in Lambert's equation
a_o	basic decrement for k_n in Lambert's equation routine (a_0 is stored in the program and cannot be modified using the load sheet format).
e	eccentricity of ellipse
I NUMBER	number of iterations required for convergence in the Lambert's equation routine
k_o	initial scaling factor for Δa in Lambert's equation routine (k_0 is stored in the program and cannot be modified using the load sheet format)
k_n	final scaling factor for Δa in Lambert's equation routine
n	integer used to scale a_o $(0 \le n \le 4)$
N	as an input, N is also used to denote the number of complete orbital circuits of the probe about the central body
OPTION	option number used in computing T_c
p	semi-latus rectum
P_i	i^{th} approximation to the period of the ellipse
Planet 1	name of the departure planet
Planet 2	name of the target planet
$\underline{r}_1$; $\underline{r}_{p1}$	heliocentric injection position; position of injection planet at T_1
$\dot{\underline{r}}_1$	heliocentric injection velocity
$\underline{r}_2$; $\underline{r}_{p2}$	heliocentric target position; position of target planet at T_2
$\dot{\underline{r}}_2$	heliocentric velocity of the probe at target position

r_{p1}	heliocentric velocity of the injection planet at injection
r_{p2}	heliocentric velocity of target planet at the arrival time
V_{hL}	hyperbolic excess speed at the injection planet
V_{hp}	hyperbolic excess speed at the target planet
$\underline{V}_{hL}$	hyperbolic excess velocity at the injection planet
$\underline{V}_{hp}$	hyperbolic excess velocity vector at the target planet
T_1	injection time expressed in Julian days in double precision
T_2	arrival time expressed in Julian days in double precision
$T_c(a_m)$	time of flight for the minimum energy trajectory
Δv	the angular distance between $\underline{r}_1$ and $\underline{r}_2$
θ_L	celestial longitude of $\underline{V}_{hL}$
θ_p	celestial longitude of $\underline{V}_{hp}$
ϕ_L	celestial latitude of $\underline{V}_{hL}$
ϕ_p	celestial latitude of $\underline{\ }_{hp}$
ϵ_T	tolerance on convergence of Lambert's equation
μ	gravitational constant of central body
$\underline{W}$	unit vector normal to the orbit plane in the sense of a right-hand screw
Δa	increment in semi-major axis in Lambert's equation iteration routine

3.3.2 Symbols for Planetocentric Phase

GM	gravitational constant of planet (in km^3/sec^2)

$\left.\begin{matrix} S_{1x} \\ S_{1y} \\ S_{1z} \end{matrix}\right\}$ components of orbit-plane determination vector, $\underline{S}_1$

$\left.\begin{matrix} W_x \\ W_y \\ W_z \end{matrix}\right\}$ components of unit vector normal to orbit plane, $\underline{W}$

B impact parameter

$\left.\begin{matrix} B_x \\ B_y \\ B_z \end{matrix}\right\}$ components of $\underline{B}$, the impact vector

$\left.\begin{matrix} T_x \\ T_y \\ T_z \end{matrix}\right\}$ components of $\underline{T}$, in the $\underline{R}$, $\underline{S}$, $\underline{T}$ system

$\left.\begin{matrix} R_x \\ R_y \\ R_z \end{matrix}\right\}$ components of $\underline{R}$ in the $\underline{R}$, $\underline{S}$, $\underline{T}$ system

$\left.\begin{matrix} \underline{B} \cdot \underline{T} \\ \underline{B} \cdot \underline{R} \end{matrix}\right\}$ parameters specifying geometry of incoming hyperbola relative to $\underline{R}$, $\underline{S}$, $\underline{T}$ system

a semi-major axis of incoming hyperbola

e eccentricity of incoming hyperbola

a_1, a_2, a_3, a_4	coefficients for polynomial used to find the optimum capture orbit radius in capture missions
P_x, P_y, P_z	components of $\underline{P}$, the perifocus vector for the incoming hyperbola
K	asymptote deflection angle
r_{opt}	radius for gravitational turns
P'_x, P'_y, P'_z	components of $\underline{P}'$, the perifocus vector for the outgoing hyperbola
r'_p	closest-approach distance on outgoing hyperbola
e'	eccentricity of outgoing hyperbola
a'	semi-major axis of outgoing hyperbola
ω	angle between perifocus vectors on incoming and outgoing hyperbolas
R_c	closest-approach distance on Earth re-entry hyperbola
r_r	radius of re-entry sphere
γ	angle between position and velocity at re-entry
φ	declination of perifocus vector at re-entry

3.4 BASIC ORGANIZATION OF THE PROGRAM

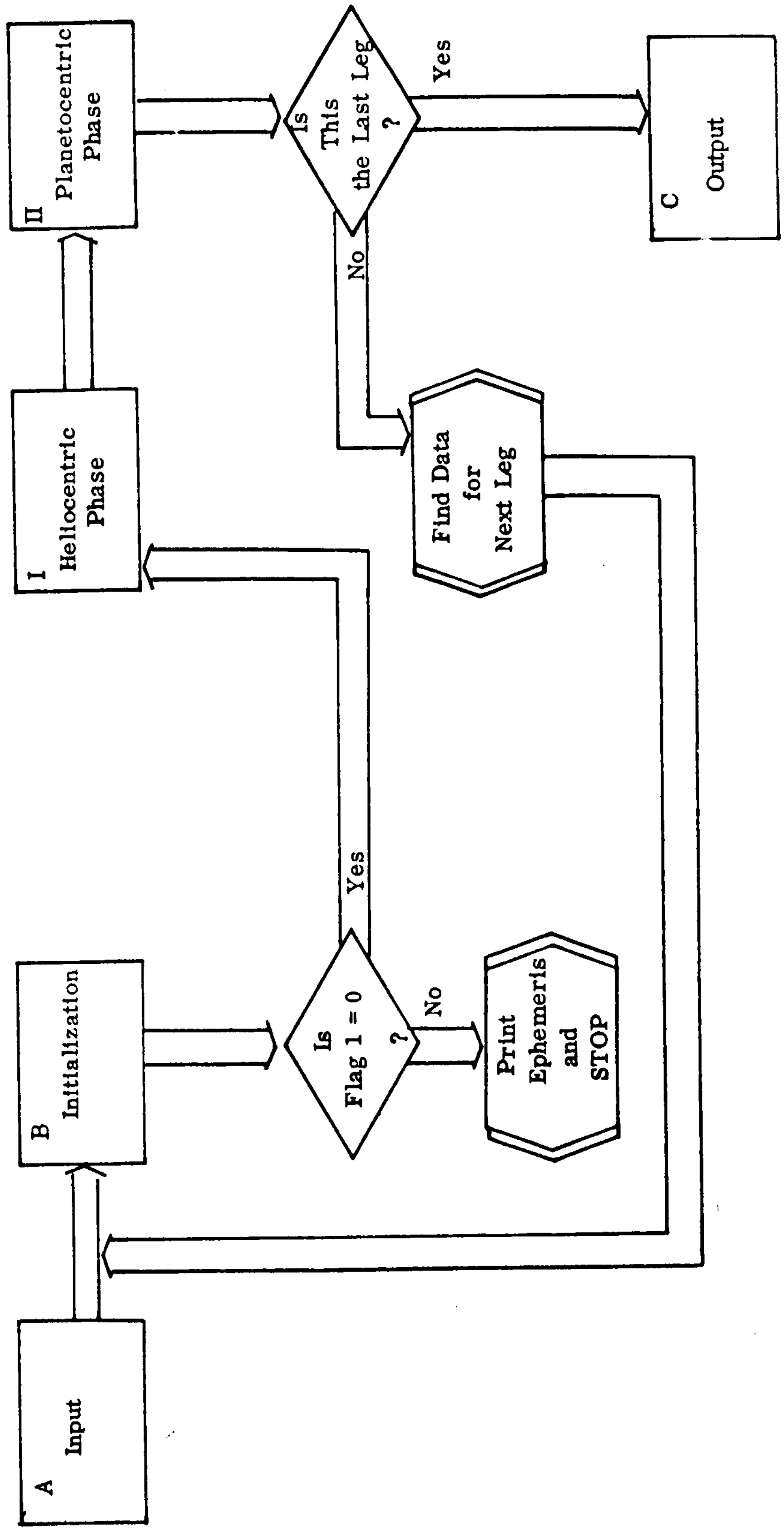
The program is divided into five basic sections or blocks as illustrated in the Level I flow chart. The INPUT block A contains data sorting and a summary of all inputs to the program that can be readily supplied or modified by the user. The INITIALIZATION block B consists of the EPHEM ephemeris routine which is described in detail in Ref. [1]. In the INITIALIZATION block, the planetary positions and velocities are computed from the dates supplied in the input. The HELIOCENTRIC block consists of the implementation of Lambert's equation to compute the semi-major axis of the heliocentric ellipse, and the computation of other elements and parameters of this ellipse. The PLANETOCENTRIC block contains the computations of parameters for the incoming and outgoing planetocentric trajectories. Finally, the OUTPUT block consists of a summary or list of all output quantities printed by the program under normal usage. It should be made clear that entry into the OUTPUT block is not necessary for printout to occur, for the program will print data groups without having entered OUTPUT. Thus the OUTPUT block in this case represents termination of the run. As the program may process several legs for a given submission, the quantities and operations described herein are assumed to be for the m^{th} leg.

4.0 INPUT, INITIALIZATION, OUTPUT

4.1 INPUT, BLOCK A

The following quantities are inputs to the program and may be easily supplied or modified by the user. Planet code names are formed using only the FIRST SIX LETTERS of the normal spelling. For example, Jupiter would be entered as JUPITE. Assembled inputs are indicated by the word "assembled" followed by the assembled value. These are values that will be used if no input is entered.

Planet 1	code name for launch planet
Planet 2	code name for target planet
Time 1	T_1 expressed in Julian days
Time 2	T_2 expressed in Julian days
r_p	pericenter distance at Planet 1, assembled, 6563.
Flag 1	option for ephemeris data only, assembled, 0
Flag 2	option for ephemeris data input, assembled, 0
Flag 3	option to input $\underline{S}_1$, assembled, 0



Level I Flow Chart

X_1, Y_1, Z_1	input values for $\underline{r}_{p1}$ (use only if Flag 2 is non-zero)
$\dot{X}_1, \dot{Y}_1, \dot{Z}_1$	input values for $\dot{\underline{r}}_{p1}$ (use only if Flag 2 is non-zero)
X_2, Y_2, Z_2	input values for $\underline{r}_{p2}$ (use only if Flag 2 is non-zero)
$\dot{X}_2, \dot{Y}_2, \dot{Z}_2$	input values for $\dot{\underline{r}}_{p2}$ (use only if Flag 2 is non-zero)
S_{1x}, S_{1y}, S_{1z}	components of $\underline{S}_1$ (use only if Flag 3 is non-zero)
μ	gravitational constant of sun (or other central body) assembled, 1.3271544×10^{11}
ϵ_T	tolerance on flight time for central body ellipse in seconds, assembled, 5000.
φ	declination of perifocus vector at Earth re-entry in decimal degrees, assembled, 28.5
γ	angle between position and velocity vectors at Earth re-entry in decimal degrees, assembled, 96.0
r_r	radius of re-entry sphere at Earth, assembled, 6500.
CENTRAL BODY	name of central body (for elliptical orbits), assembled, SUN
N	N, the number of <u>complete</u> circuits encompassing the central body in time interval $T_2 - T_1$, assembled, 0. Note: If $N > 0$, the approximate value of the semi-major axis of the desired ellipse must be specified.

Position and velocity must be expressed in km. and km./sec., respectively, while μ is given in km^3/sec^2. The gravitational constants for the planets are stored internally, and will be called whenever the corresponding planet is entered.

4.2 INITIALIZATION - BLOCK B, THE EPHEMERIS ROUTINE AND ASSOCIATED LOGIC

As the EPHEM ephemeris routine is described in detail elsewhere [2] only its general aspects will be discussed here.

The routine contains a tabulation of positions and velocities for all nine planets and the moon from December 30, 1949 (Julian date, 2433280.5) to January 5, 2000 (Julian date 2451548.5). The data is recorded on three computer tapes with overlapping ranges. The tapes are divided as follows:

TAPE	JULIAN DATE (CALENDAR DATE)	TO	JULIAN DATE (CALENDAR DATE)
EPHEM 1	2433280.5 (Dec. 30, 1949)		2440584.5 (Dec. 29, 1969)
EPHEM 2	2439500.5 (Jan. 10, 1969)		2446796.5 (Jan. 1, 1987)
EPHEM 3	2445708.5 (Jan. 9, 1984		2451548.5 (Jan. 5, 2000)

The data is tabulated in .5 day intervals for the Moon, 2 day intervals for Mercury, and 4 day intervals for the remaining planets. The corresponding second and fourth differences are also recorded. Position and velocity for a particular planet at a particular time is computed from Everett's interpolation formula utilizing second and fourth differences.

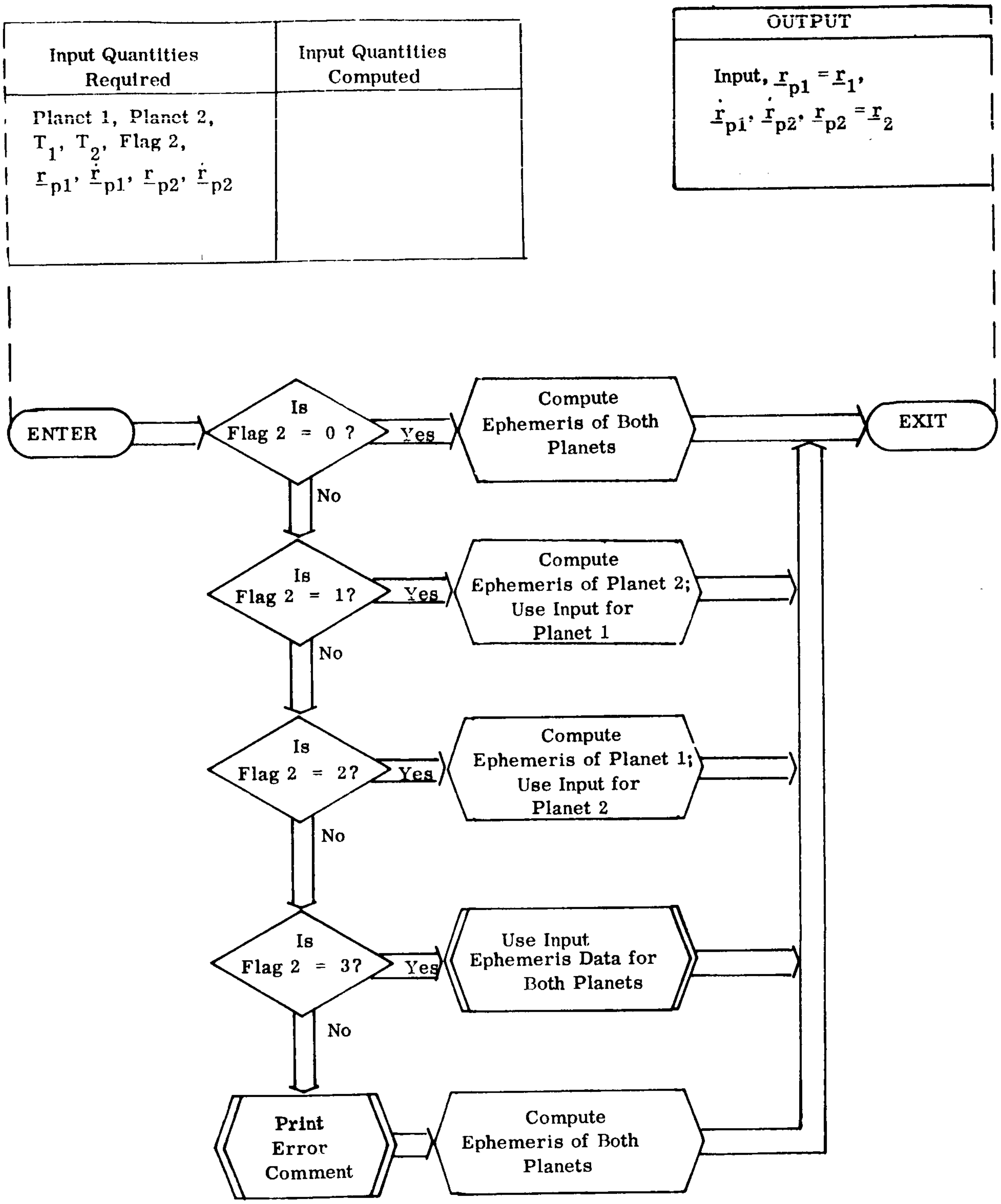
A Level II flow chart of the Initialization block is shown.

4.3 OUTPUT BLOCK C

The output quantities consist of input and those quantities computed by the program. As the input has been listed previously, only the computed quantities will be given here with some overlap for those which could be either computed or input. A sample of computer print is shown in the check case in Section 7.0.

4.3.1 Heliocentric Phase

X*P1, Y*P1, Z*P1	components of $\underline{r}_{p1}$
XDOT*P1, YDOT*P1, ZDOT*P1	components of $\dot{\underline{r}}_{p1}$
X*P2, Y*P2, Z*P2	components of $\underline{r}_{p2}$
XDOT*P2, YDOT*P2, ZDOT*P2	components of $\dot{\underline{r}}_{p2}$
V*HLX, V*HLY, V*HLZ	cartesian components of the $\underline{V}_h$ vector leaving planet 1 (V_{hLX}, V_{hLY}, V_{hLZ})



Level II Flow Chart - Initialization Block

V*HL	magnitude of the vector above ($\underline{V}_{hL}$)
THETA*L	right ascension of the vector above (θ_L)
PHI*L	declination of the vector above (ϕ_L)
V*HPX, V*HPY, V*HPZ	cartesian components of the $\underline{V}_h$ vector arriving at Planet 2 (V_{hPX}, V_{hPY}, V_{hPZ})
V*HP	magnitude of the vector above ($\underline{V}_{hp}$)
THETA*P	right ascension of the vector above (θ_P)
PHI*P	declination of the vector above (ϕ_P)
P	p
E	e
A	a
OPTION	number of particular form used for Lambert's equation (1, 2, 3, 4)
DELTA V	ΔV
I NUMBER	number of iterations in Lambert's equation
W*X, W*Y, W*Z	components of $\overline{W}$
T*CM	T_{cm}
K*O	k_o
K*N	k_n
A*O	a_o
EP*T	ϵ_T
MU	μ
CENT BODY	central body

FLAG 1	flag 1
FLAG 2	flag 2
FLAG 3	flag 3
X*1	X_1 (input)
Y*1	Y_1 (input)
Z*1	Z_1 (input)
XDOT*1	$\dot{X}$ (input)
YDOT*1	$\dot{Y}$ (input)
ZDOT*1	$\dot{Z}$ (input)
V*HPXP V*HPYP V*HPZP	components of $\underline{V}'_{hp}$
EN	N
AIN	a_1 (initial value of semi-major axis in Lambert's equation)

4.3.2 Planetocentric Phase, Planet 1

GRAV PLANET 1	gravitational constant of planet 1
GRAV PLANET 2	gravitational constant of planet 2
GRAV CNTRL BODY	μ
R*P	r_p
S*1X S*1Y S*1Z	components of $\underline{S}_1$
W*X W*Y W*Z	components of $\underline{W}$

B	B
B*X, B*Y, B*Z	components of $\underline{B}$
T*X, T*Y, T*Z	components of $\underline{T}$
R*X, R*Y, R*Z	components of $\underline{R}$
B.T	$\underline{B} \cdot \underline{T}$
B.R	$\underline{B} \cdot \underline{R}$
A	a
E	e
A*1, A*2, A*3, A*4	a_1, a_2, a_3, a_4
P*X, P*Y, P*Z	components of $\underline{P}$
K	K
R*OPT	r_{opt}
P'*X, P'*Y, P'*Z	components of $\underline{P}'$
R'*P	r'_p
E'	e'
A'	a'
OMEGA	ω

4.3.3 Planetocentric Section, Re-Entry

A	a
E	e
R*C	R_c
THETA	θ (right ascension of perifocus vector)

W*X, W*Y, W*Z	components of $\underline{W}$
B*X, B*Y, B*Z	components of $\underline{B}$
B	B
T*X, T*Y, T*Z	components of $\underline{T}$
R*X, R*Y, R*Z	components of $\underline{R}$
R*R	r_r
B.T	$\underline{B} \cdot \underline{T}$
B.R	$\underline{B} \cdot \underline{R}$
GAMMA	γ
PHI	φ

5.0 BASIC COMPUTATION BLOCKS

5.1 BLOCK I - HELIOCENTRIC PHASE

The Heliocentric Phase consists of two basic routines, Lambert's equation and the Conic Determination. All of the quantities computed in this section utilize the inputs $\underline{r}_{p1}$, $\underline{r}_{p2}$, $\dot{\underline{r}}_{p1}$, $\dot{\underline{r}}_{p2}$, T_1, T_2 as obtained from the input and initialization block.

It is possible for Lambert's equation not to converge; this will occur, for example, if the resulting ellipse has too large an eccentricity. Should convergence fail, the program will stop. Also, in the Heliocentric Conic Determination section, if $\underline{r}_{p1}$ and $\underline{r}_{p2}$ are collinear or nearly so, the orbit plane defined by $\underline{W}'$ cannot be computed uniquely. If this situation arises, the program will print the comment "VECTORS ARE NEARLY COLLINEAR" and stop.

5.1.1 Block I.1 - Lambert's Equation

INPUT: $\underline{r}_{p1} = \underline{r}_1$, $\underline{r}_{p2} = \underline{r}_2$, μ, T_1, T_2, ϵ_T, k_o, a_o

OUTPUT: a, $\underline{W}'$

1. $T = 86400. (T_2 - T_1)$

2. $$\underline{W}' = \frac{\underline{r}_1 \times \underline{r}_2}{|\underline{r}_1 \times \underline{r}_2|}$$

3. $$c = |r_2 - r_1|$$

4. $$s = \frac{r_1 + r_2 + c}{2}$$

Iterative Portion (i = 1, 2, 3, . . . , 15)

5. $$a_1 = \frac{s}{2}$$

6. $$\cos \alpha_i = 1 - \frac{s}{a_i}$$

where

$$\cos \alpha_1 = -1.0$$

7. $$\cos \beta_i = 1 - \frac{s - c}{a_i}$$

8. $$\alpha_i = \cos^{-1} \cos \alpha_i \qquad (0 \leq \alpha_i \leq \pi)$$

9. $$\beta_i = \cos^{-1} \cos \beta_i \qquad (0 \leq \beta_i \leq \pi)$$

10. $$\sin \alpha_i = \sqrt{1 - \cos^2 \alpha_i}$$

11. $$\sin \beta_i = \sqrt{1 - \cos^2 \beta_i}$$

12. $P_i = 2\pi \sqrt{\frac{a_i^3}{\mu}}$

13.

Option 1: $T_{ci} = \frac{P_i}{2\pi} [(\alpha - \sin \alpha) - (\beta - \sin \beta)] + NP_i$

Option 2: $T_{ci} = \frac{P_i}{2\pi} [(\alpha - \sin \alpha) + (T - \sin \beta)] + NP_i$

Option 3: $T_{ci} = P_i(N + 1) - \frac{P_i}{2\pi} [(\alpha - \sin \alpha) + (\beta - \sin \beta)]$

Option 4: $T_{ci} = P_i(N + 1) - \frac{P_i}{2\pi} [(\alpha - \sin \alpha) - (\beta - \sin \beta)]$

14. $\Delta T_i = T - T_{ci}$

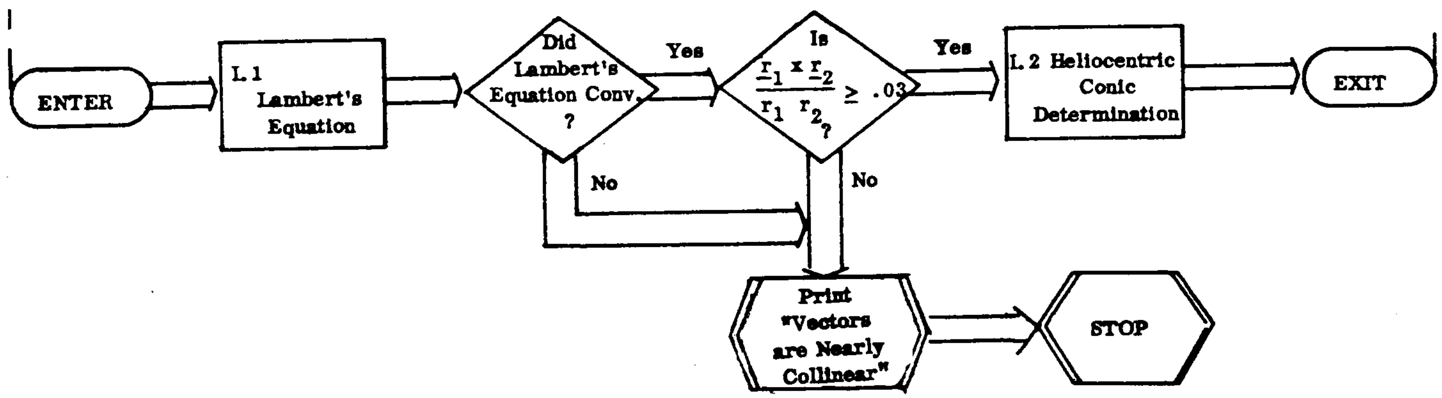
15. $a_{i+1} = a_i + \Delta a_i$

$$\Delta a_i = k_n \frac{a_i - a_{i-1}}{T_{ci} - T_{ci-1}} (T - T_{ci}) \qquad i = 1, 2, 3, \ldots, 15$$

Note: i is set to zero on one occasion (see flow chart)

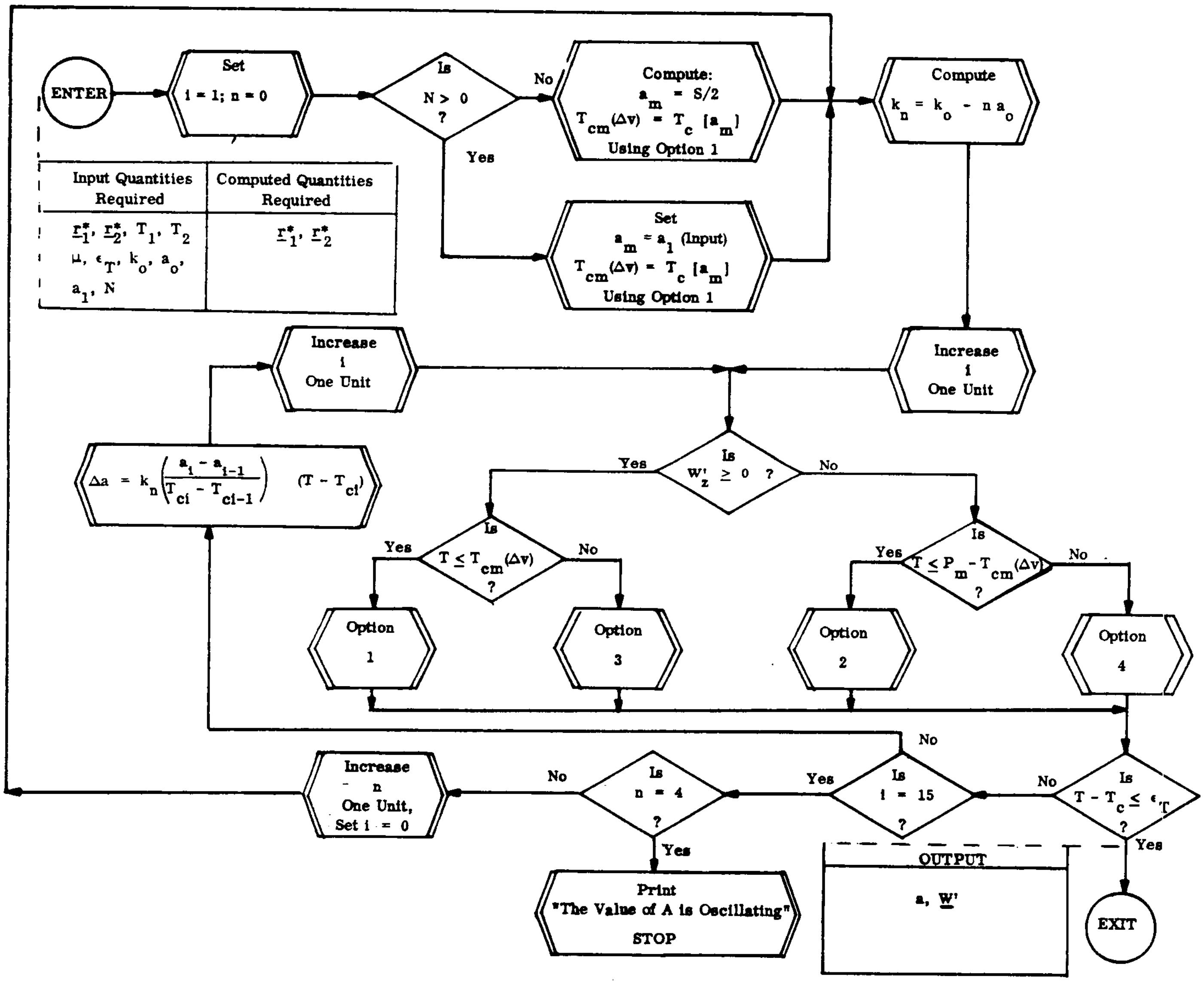
Input Quantities Required	Computed Quantities Required
T_1, T_2, ϵ_T, a_1, N, a_o, k_D, μ, $\underline{r}^*_1$, $\underline{r}^*_2$, $\dot{\underline{r}}^*_{p1}$, $\dot{\underline{r}}^*_{p2}$	$\underline{r}^*_1$, $\underline{r}^*_2$, $\dot{\underline{r}}^*_{p1}$, $\dot{\underline{r}}^*_{p2}$

OUTPUT
a, p, e, $\dot{\underline{r}}_{p1}$, $\dot{\underline{r}}_{p2}$, $\underline{V}_{hL}$, $\underline{V}_{hp}$, V_{hL}, V_{hp}, θ_L, ϕ_L, ϕ_p, θ_p, μ, $\dot{\underline{r}}_1$, $\dot{\underline{r}}_2$, $\underline{W}$, OPTION, T_{cm}, Δv, I Number, k_o, a_o



* These quantities may be computed or input, depending upon the state of Flag 2

Level II Flow Chart - Heliocentric Phase



*$\underline{r}_1$ and $\underline{r}_2$ may be input or computed, depending on the state of Flag 2

Level III Flow Chart - Lambert's Equation

5.1.2 Block I.2 - Heliocentric Conic Determination

INPUT: a, $\underline{W}'$

OUTPUT: $p, e, r_1, r_2, P, \underline{V}_{hL}, \underline{V}_{hp}, \phi_L, \theta_L, \phi_p, \theta_p$

$$p = [4a(s - r_1)(s - r_2)/c^2] \sin^2 [(\alpha \pm \beta)/2]$$

- sign used if option 2 or 3 was used

+ sign used if option 1 or 4 was used

1.

a. $e \cos v_1 = \dfrac{p}{r_1} - 1$

b. $e \cos v_2 = \dfrac{p}{r_2} - 1$

2.

a. $\cos \Delta v = \dfrac{\underline{r}_1 \cdot \underline{r}_2}{r_1 \cdot r_2}$

b. $\sin \Delta v = \pm \sqrt{1 - (\cos \Delta v)^2}$

c. Is $W'_z \geq 0$

Yes: Sign of $(\sin \Delta v) \geq 0$; $\underline{W} = \underline{W}'$

No: Sign of $(\sin \Delta v) < 0$; $\underline{W} = -\underline{W}'$

3. $e \sin v_1 = \dfrac{(\cos \Delta v)(e \cos v_1) - (e \cos v_2)}{(\sin \Delta v)}$

4. $e = + \sqrt{(e \sin v_1)^2 + (e \cos v_1)^2}$

5. $\dot{r}_1 = \dfrac{(\sqrt{\mu})(e \sin v_1)}{\sqrt{p}}$

6. $\underline{U}_{V1} = \underline{W} \times \underline{r}_1 / r_1$

7. $\underline{V}_1 = \dot{\underline{r}}_1 = (\dot{r}_1)\left(\frac{\underline{r}_1}{r_1}\right) + \frac{\sqrt{\mu}\ \sqrt{p}}{(r_1)}\ \underline{U}_{V1}$

8. $e \sin v_2 = \frac{(e \cos v_1) - (\cos \Delta v)(e \cos v_2)}{\sin \Delta v}$

9. $\dot{r}_2 = \frac{\sqrt{\mu}\ (e \sin v_2)}{\sqrt{p}}$

10. $\underline{U}_{V2} = \underline{W} \times \frac{\underline{r}_2}{r_2}$

11. $\underline{V}_2 = \dot{\underline{r}}_2 = (\dot{r}_2)\left(\frac{\underline{r}_2}{r_2}\right) + \frac{\sqrt{\mu}\ \sqrt{p}}{r_2}\ \underline{U}_{V2}$

12. $\underline{P} = \frac{(e \cos v_1)}{(e)}\ \frac{\underline{r}_1}{r_1} - \frac{(e \sin v_1)}{(e)}\ \underline{U}_{V1}$

13. $\underline{V}_{hL} = \underline{V}_1 - \dot{\underline{r}}_{p1}$

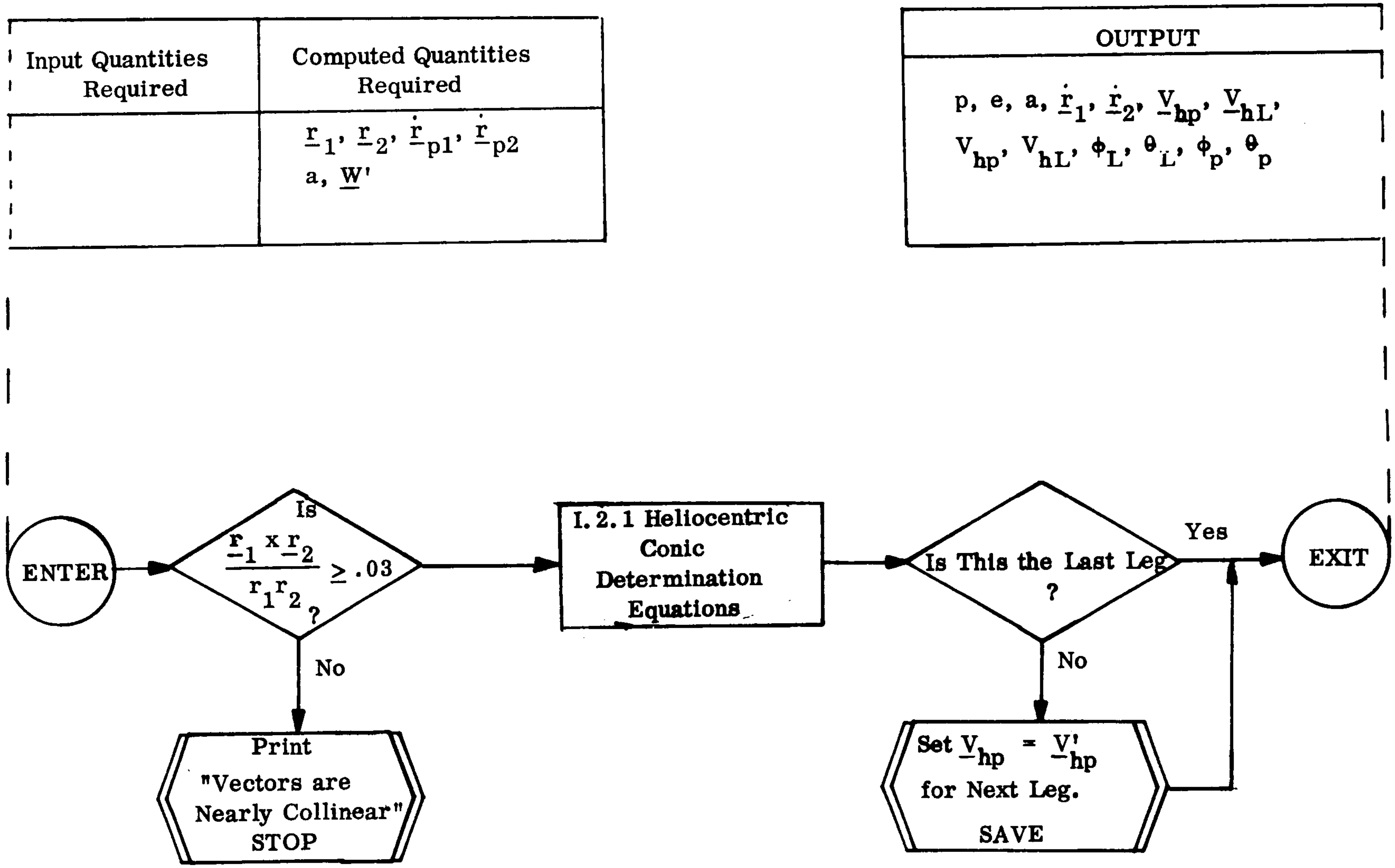
14. $\underline{V}_{hp} = \underline{V}_2 - \dot{\underline{r}}_{p2}$

15. $\phi_L = \sin^{-1} (V_{hLZ}/V_{hL}$ $\qquad -\pi/2 \le \phi_L \le \pi/2$

16. $\theta_L = \tan^{-1} \left(\frac{V_{hLY}}{V_{hLX}}\right)$ $\qquad 0 \le \theta_L < 2\pi$

17. $\phi_p = \sin^{-1} (V_{hpZ}/V_{hp})$ $\qquad -\pi/2 \le \phi_p \le \pi/2$

18. $\theta_p = \tan^{-1} \left(\frac{V_{hpY}}{V_{hpX}}\right)$ $\qquad 0 \le \theta_p < 2\pi$



Level III Flow Chart - Heliocentric Conic Determination

5.2 PLANETOCENTRIC PHASE - COMPUTATION BLOCK II

The planetocentric equations involve computation of the incoming and outgoing hyperbolas at the planets in question. Normally, the computations are made for Planet 1 unless Planet 2 is EARTH, in which case, an additional set of equations (re-entry) will be computed for this planet. The computations are conveniently divided into three blocks, incoming trajectory, outgoing trajectory, and re-entry (see flowchart on page 40).

5.2.1 Incoming Trajectory, Block II.1

Equations

INPUT: $\underline{V}_{hL}$, $\underline{V}'_{hp}$, r_p, GM, $\underline{S}_1$

OUTPUT: $\underline{S}_1$, $\underline{W}$, B, $\underline{B}$, $\underline{R}$, $\underline{T}$, a, e, a', e', a_1, a_2, a_3, a_4, $\underline{P}$, K, r_{opt}, r_p, $\underline{B}\cdot\underline{T}$, $\underline{B}\cdot\underline{R}$

Re-entry

a, e, R_c, θ, $\underline{W}$, $\underline{B}$, B, $\underline{T}$, $\underline{R}$, r_r, $\underline{B}\cdot\underline{T}$, $\underline{B}\cdot\underline{R}$, γ, φ, $\underline{V}'_{hp}$

1. (for planet 1)

$$\underline{W} = \frac{\underline{S}_1 \times \underline{S}_2}{|\underline{S}_1 \times \underline{S}_2|} \quad (\underline{W}, \underline{S}_1 \text{ are printed})$$

where

$$\underline{S}_1 = \frac{\underline{V}'_{hp}}{|\underline{V}'_{hp}|}$$

If $|\underline{V}'_{hp}| = 0$, set $\underline{S}_1 = 0,0,0$, set $|\underline{V}'_{hp}| = |\underline{V}_{hL}|$

$$\underline{S}_2 = \frac{\underline{V}_{hL}}{|\underline{V}_{hL}|}$$

If Flag 3 is out, and Planet 1 is EARTH,

$$\underline{S}_1 = \begin{bmatrix} \cos(\theta_L - \pi/2) \\ \sin(\theta_L - \pi/2) \\ 0 \end{bmatrix}$$

2. a.

$$B = \sqrt{\frac{2\,GM\,r_P}{|\underline{V}'_{hp}|^2} + r_P^{\,2}}$$

b.

$$\underline{1}_B = \underline{S}_1 \times \underline{W}$$

c.

$$\underline{B} = B\,\underline{1}_B$$ ($\underline{B}$ is printed)

3.

$$\underline{T} = \begin{bmatrix} \dfrac{S_{1Y}}{\sqrt{S_{1X}^2 + S_{1Y}^2}} \\ \dfrac{-S_{1X}}{\sqrt{S_{1X}^2 + S_{1Y}^2}} \\ 0 \end{bmatrix}$$ ($\underline{T}$ is printed)

4.

$$\underline{R} = \underline{S}_1 \times \underline{T}$$ ($\underline{R}$ is printed)

5.

$\underline{B} \cdot \underline{T}$ and $\underline{B} \cdot \underline{R}$ are computed and printed

6. a.

$$a = -\frac{GM}{|\underline{V}'_{hp}|^2}$$ (a is printed)

b. $$e = 1 - \frac{r_p}{a}$$ (e is printed)

7. Let

$$C_1 = \frac{|\underline{V}'_{hp}|^2}{GM} \quad , \quad C_2 = \frac{|\underline{V}_{hL}|^2}{GM}$$

a_1, a_2, a_3, a_4 are computed and printed, where

$$a_1 = 8\,(C_1 + C_2)$$

$$a_2 = -\,(C_1 - C_2)^2$$

$$a_3 = -2\,(C_1^2\,C_2 + C_1\,C_2^2)$$

$$a_4 = -C_1^2\,C_2^2$$

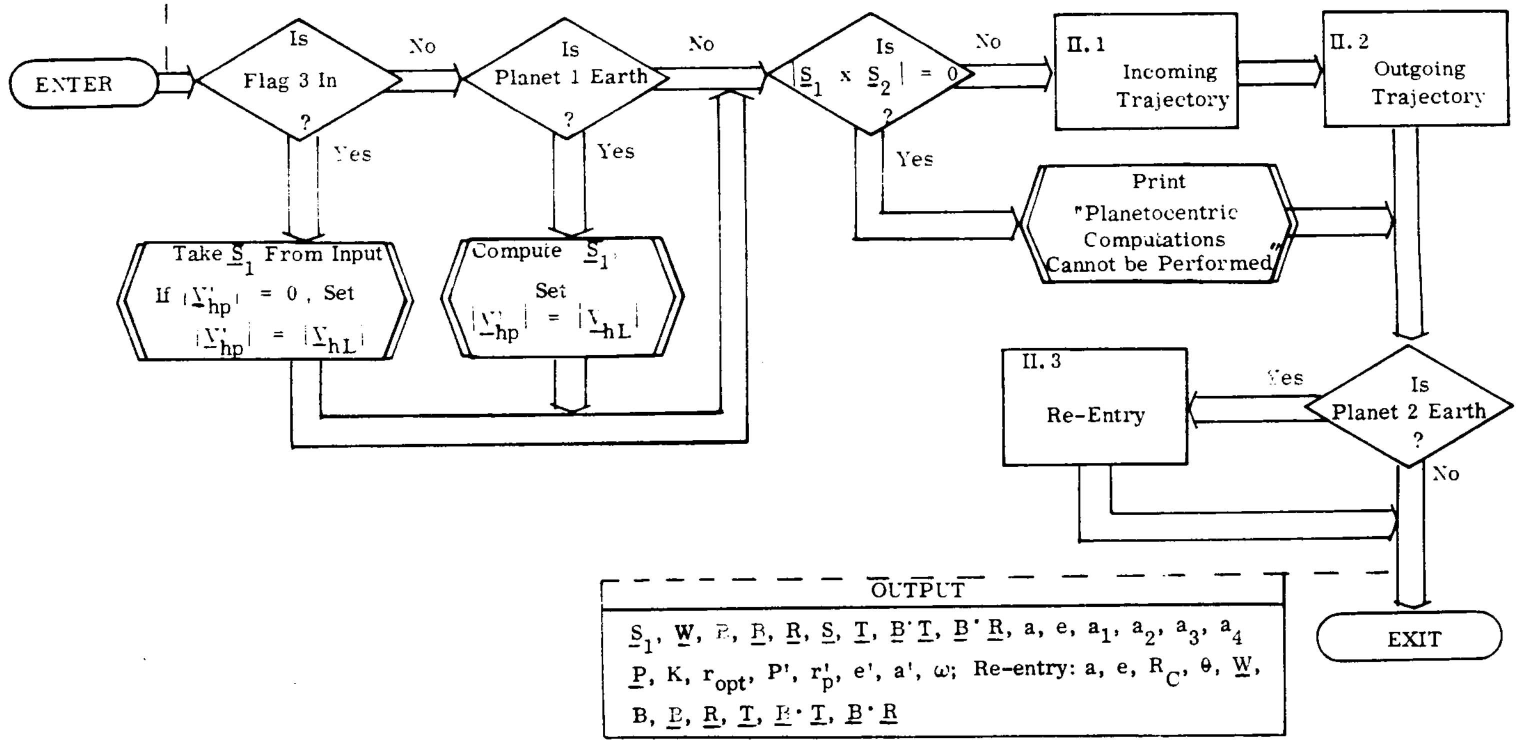
8. $$K = \cos^{-1} \underline{S}_1 \cdot \underline{S}_2$$ (1st or 2nd quadrants; K is printed in decimal degrees)

9. $$r_{opt} = \frac{GM}{|\underline{V}'_{hp}|^2} \left(\csc \frac{K}{2} - 1\right)$$ (r_{opt} is printed)

10. $$\underline{S}_3 = \underline{S}_1 \times \underline{W}$$

11. $$\underline{P} = \sqrt{\frac{e^2 - 1}{e^2}}\;\underline{S}_3 + \frac{1}{e}\;\underline{S}_1$$ ($\underline{P}$ is printed)

Input Quantities Required	Computed Quantities Required
Flag 3, GM, Planet 1, Planet 2, r_p, $\underline{S}^*_1$, Φ, γ, r_r	$\underline{V}_{hp}$, $\underline{V}_{hL}$, $\underline{V}'_{hp}$, $\underline{S}^*_1$



* $\underline{S}_1$ may be computed or input, depending on the state of Flag 3

Level II Flow Chart, Computation Block II, Planetocentric Phase

5.2.2 Outgoing Trajectory, Block II.2

Equations

INPUT: $\underline{W}$, V_{hL}, GM, r_p

OUTPUT: a', e', r'_p, ω, $\underline{P}'$

1. a.
$$S_u = -\underline{S}_2 \cdot \underline{P}$$

b.
$$S_v = -\underline{S}_2 \cdot (\underline{P} \times \underline{W})$$

2.
$$a' = -\frac{GM}{|\underline{V}_{hL}|^2}$$
(a' is printed)

3.
$$Z^2 = \frac{r_p^2 S_V^2 + r_p S_V \sqrt{r_p^2 S_v^2 - 4a' r_p (S_u + 1)} - 2a' r_p (S_u + 1)}{2a'^2}$$

4.
$$e' = \sqrt{Z^2 + 1}$$
(e' is printed)

5.
$$r'_p = a' (1 - e')$$
(r'_p is printed)

6.
$$\cos \omega = \frac{a' (1 - e'^2) - r_p}{e' r_p}$$

7.
ω is computed and printed (1st and 2nd quadrant)

8.
$$\underline{P}' = \frac{S_u \cos \omega - \frac{1}{e'}}{S_V^2} \underline{S}_2 + \frac{\cos \omega - \frac{S_u}{e'}}{S_V^2} \underline{P}$$
($\underline{P}$ is printed)

9. Test

$$\left| \frac{\underline{P}' \times \underline{P}}{\left|\underline{P}' \times \underline{P}\right|} - \underline{W} \right| \; : \; 0$$

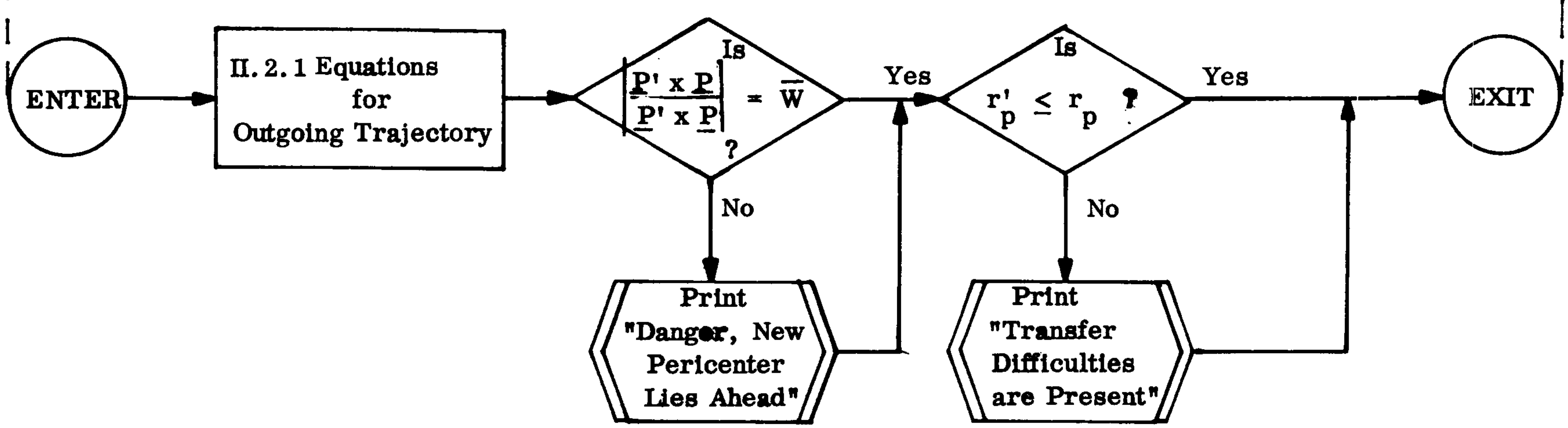
If an inequality results, the comment, "DANGER, NEW PERICENTER LIES AHEAD", is printed.

10. A test is made. If

$r'_p > r_p$, the comment, "TRANSFER DIFFICULTIES ARE PRESENT", is printed.

Input Quantities Required	Computed Quantities Required
GM	$\underline{S}_1$, $\underline{V}_{hL}$, $\underline{V}_{hp}$, $\underline{W}$ r_p

OUTPUT
$\underline{P}'$, a', e', r'_p, ω



Level III Flow Chart - Outgoing Trajectory

5.2.3 Block II. 3, Re-Entry

Equations

INPUT: φ, γ, $\underline{V}_{hp}$, GM

OUTPUT: a, e, R_c, θ, $\underline{W}$, $\underline{B}$, B, $\underline{B}\cdot\underline{T}$, $\underline{B}\cdot\underline{R}$, $\underline{R}$, $\underline{T}$

1. $$a = -\frac{GM}{|\underline{V}_{hp}|^2}$$

2.

a. $$e = \sqrt{1 + \frac{r_r \sin^2 \gamma}{|a|^2} (2|a| + r_r)}$$ (e is printed)

b. $$R_C = a(1 - e)$$ (R_C is printed)

3.

a. Let $\underline{S} = \dfrac{\underline{V}_{hp}}{|\underline{V}_{hp}|}$. Then

$$C = \frac{1}{\cos \phi} (S_Z \sin \phi - 1/e)$$

b.

$$\sin \theta = \frac{- C S_Y + n S_X \sqrt{S_X^2 + S_Y^2 - C^2}}{S_X^2 + S_Y^2}$$

$$\cos \theta = \frac{- C S_X + m S_Y \sqrt{S_X^2 + S_Y^2 - C^2}}{S_X^2 + S_Y^2}$$

$m = - n$; $|n| = |m| = 1$. On the first pass $n = 1$

(1) If $S_X^2 + S_Y^2 - C^2 < 0$, a comment is printed,

"RE-ENTRY CONDITIONS CANNOT BE OBTAINED".

(2) If $S_X^2 + S_Y^2 = 0$, a comment is printed,

"JOB ABORTED DUE TO SINGULAR POLAR ORBIT".

Either of the two conditions above shall terminate the case.

c.

$$W_Z = \frac{e \cos \phi}{\sqrt{e^2 - 1}} (S_Y \cos \theta - S_X \sin \theta) \qquad \text{(save } W_Z\text{)}$$

d. W_Z is tested against zero. If $W_Z > 0$, θ is computed and printed in decimal degrees, and the next step will be equation (4). If $W_Z < 0$, n is set equal to -1; sin θ, cos θ are recomputed and printed in decimal degrees, and the next step will be equation (4).

4. a.

$$W_X = \frac{e}{\sqrt{e^2 - 1}} (S_Z \cos \phi \sin \theta - S_Y \sin \phi)$$

$$W_Y = \frac{e}{\sqrt{e^2 - 1}} (S_X \sin \phi - S_Z \cos \phi \cos \theta)$$

$$W_Z = \frac{e \cos \phi}{\sqrt{e^2 - 1}} (S_Y \cos \theta - S_X \sin \theta)$$

5.

a.

$$\overline{1}_B = \frac{\overline{S} \times \overline{W}}{|\overline{S} \times \overline{W}|}$$

b.

$$B = \sqrt{\frac{2 \text{ GME } R_C}{|V_{hp}|^2} + R_C^2}$$

c. $\underline{B} = B \, \underline{1}_B$ ($\underline{B}$ is printed)

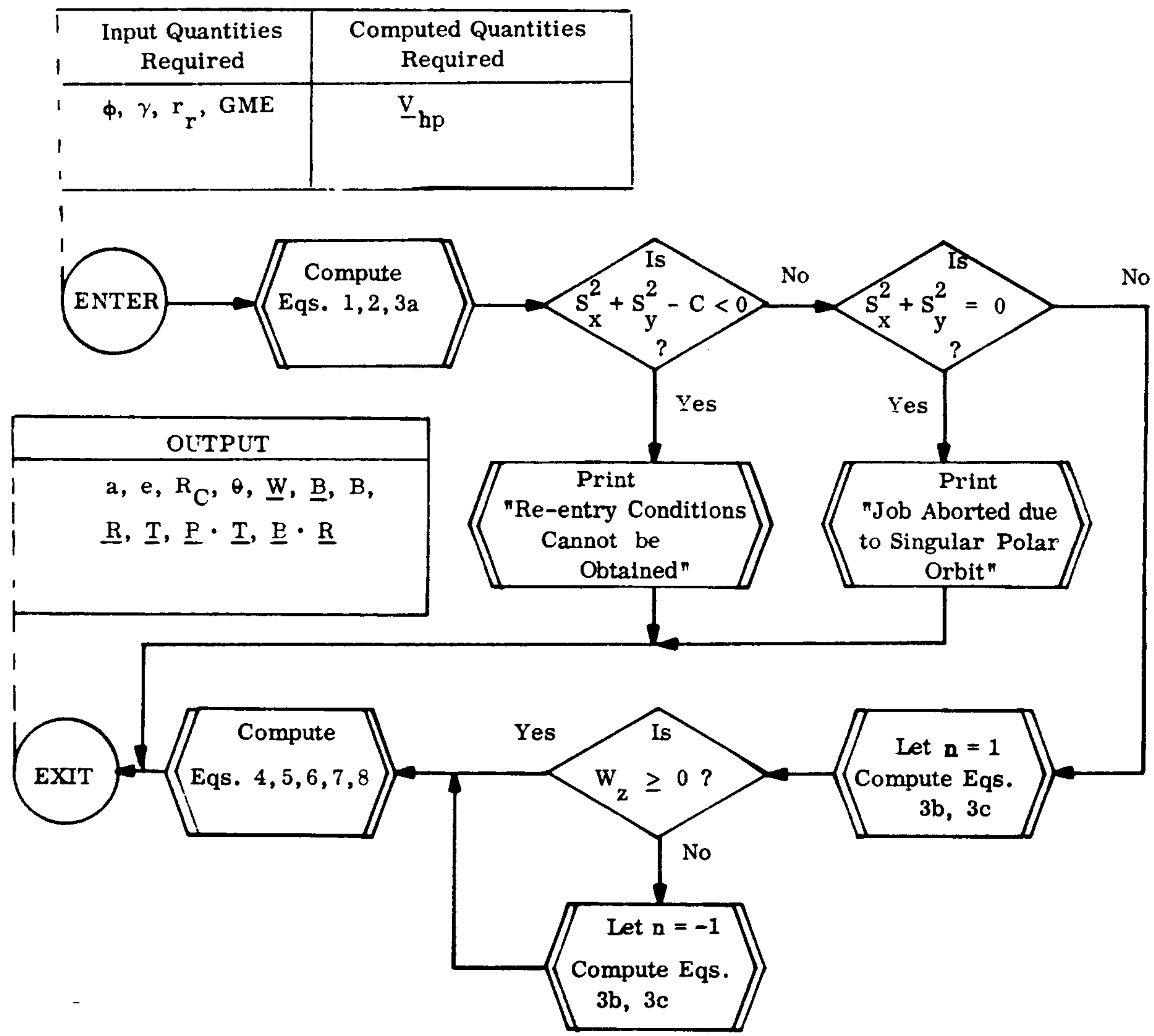
6.

$$\underline{T} = \begin{bmatrix} \dfrac{S_Y}{\sqrt{S_X^2 + S_Y^2}} \\ \\ \dfrac{-S_X}{\sqrt{S_X^2 + S_Y^2}} \\ \\ 0 \end{bmatrix}$$

($\underline{T}$ is printed)

7. $\underline{R} = \underline{S} \times \underline{T}$ ($\underline{R}$ is printed)

8. $\underline{B} \cdot \underline{T}$ and $\underline{B} \cdot \underline{R}$ are computed and printed.



Level III Flow Chart - Re-Entry

6.0 USER'S GUIDE

This section contains a description of the various possibilities which are provided by the Program 291.1 for the determination of an interplanetary free fall mission profile. In addition, a description of the role of Program Deck 291.1 in the conic matching procedure is given.

The description of the use of Program 291.1 is subdivided into two sections. In Paragraph 6.1 the program options as determined by the flags are explained independently from the particular mission design. In Paragraph 6.2 the applications of the program to particular missions are described and a sample case is provided.

A sample case is shown following the loadsheets to which it corresponds. The resulting computer print shows the output quantities which are defined in Paragraphs 3.3 and 4.3.

Although it is not claimed that the list of applications of Program 291.1, as shown, exhausts all possibilities, the applications given do, in fact, cover many situations commonly encountered in the definition of interplanetary missions.

6.1 FLAGS AND THEIR USE WITH ASSOCIATED OPTIONS

Three flags (numbers 1, 2, 3) are provided, two of which relate to the heliocentric phase and one of which relates to the planetocentric phase. These flags allow the user to realize various schemes of computation. All flags have the value zero assembled into the program. This value will be used if no data are entered into the corresponding location.

6.1.1 Heliocentric Phase

The Heliocentric Phase contains two flags, numbered 1 and 2, respectively. Flag 1, if set to a non-zero value, will eliminate all computations except for the ephemeris of the two designated planets. Its values may be specified as zero or non-zero. If the value is set equal to zero, all computation according to specified input will be performed.

Flag 2 can take on the value 0, 1, 2, 3 and provides additional flexibility in terms of ephemeris input to Lambert's equation. If the value is zero, the ephemeris of the planets specified at the input sheet will be provided. If any value other than 1, 2 or 3 is entered, the program will assume the value zero and print an error comment. The ephemeris will then be computed from values taken from the tape.

Flag 2 is set to a non-zero value whenever it is desired to supply to the Lambert's equation routine position vectors other than those defined by the heliocentric (or planetocentric) positions of the planets as specified by tape data. For applications see Paragraph 6.2.5 (Flag 2 = 1), Paragraphs 6.2.2 and 6.2.3 (Flag 2 = 2), Paragraph 6.2.4 and Paragraph 6.2.6 (Flag 2 = 3).

The ephemeris of the planets with respect to bodies other than the sun may be obtained by placing the code name of that body in the CENTRAL BODY input. This will cause a shift in coordinates from a sun-centered to a planet-centered system. Such a change is accomplished by a translation, and the orientation of the coordinate system remains the same, however. Such a technique may be used to obtain communication distances and data associated with tracking.

Circular and elliptical orbits, with respect to a planet rather than the Sun, may be computed given the flight time (expressed by $T_2 - T_1$) and two positions $\underline{r}_1$, $\underline{r}_2$ through use of the Flag 2 input and a change of the gravitational constant of the central body. This is explained in greater detail in Paragraph 6.2.

6.1.2 Planetocentric Phase

Program 291.1 computes certain orbital parameters for the planetocentric conics. Flag 3 controls the specification of the planetocentric orbit plane for planetary departure. Two cases have to be distinguished:

$$\text{Flag } 3 = 0$$

$$\text{Flag } 3 \neq 0$$

6.1.2.1 Flag 3 = 0

In normal usage, Flag 3 assumes its assembled value of zero. In such cases, the first planet of the first leg is EARTH, and orbit planes for departure planets in subsequent legs are determined from the incoming and outgoing $\underline{V}_h$ vectors. $\underline{S}_1$ for Earth departure is computed on the criterion of maximum orbit plane inclination.

6.1.2.2 Flag 3 ≠ 0

In this case the orbit plane is specified by the outgoing hyperbolic excess vector ($\underline{V}_{hL}$) and a vector $\underline{S}_1$, which must be input.

6.2 APPLICATIONS TO SPECIFIC PROBLEMS

Problems which frequently arise in the determination of interplanetary missions are listed below. In each case, the corresponding use of Program 291.1 is explained.

6.2.1 Round-trip Flyby Trajectories

A round-trip mission with all flags set at zero constitutes normal usage of the program. In turn, a round-trip mission will be defined as any interplanetary mission which departs from Earth, encounters at least one target planet along the path, and terminates at the Earth. Any number of target planets may be incompased in such a case; this includes missions of the grand tour and swingby types. In such normal usage, the data for each leg is stacked sequentially. Stopover missions which do not enter the planetary atmosphere are also included in this list, the only difference being that arising from the difference in arrival and departure times due to the stopover interval.

6.2.2 One-way Missions

One-way missions are those interplanetary missions which neither encompass more than one target planet, nor return to Earth. These may include probe flybys or planetary landing missions. The computation procedure is as follows:

1. The Earth-departure and target planet arrival times (T_1 and T_2, respectively) are specified. This information, together with the planetary code names, is entered as data for the desired leg.

2. A third time T_3 ($T_3 > T_2$) and corresponding heliocentric position are computed.

3. A second or "pseudo" leg is formed, using as input times T_2 and T_3, Flag 2 = 2, and the computed heliocentric position vector entered as X_2, Y_2, Z_2. Any planet code name may be entered for Planet 2 in this leg. The resulting computations will describe the trajectory parameters at the desired target planet.

6.2.3 Atmospheric Entry at a Planet Other than Earth

If it is desired to simulate atmospheric entry for a planet other than Earth, the following steps must be taken:

1. Set Flag 2 = 2 and input the ephemeris ($\underline{r}_{p2}$, $\dot{\underline{r}}_{p2}$) of the desired planet.

2. Use for PLANET 2 the code name EARTH

3. Enter the gravitational constant of the desired planet for GME

4. Enter the desired atmospheric entry parameters r_r, ϕ, γ.

Input data for the departure planet is not affected by the steps above. Atmospheric breaking and planetary landing may be handled in this manner.

6.2.4 Computation of Elliptical Orbits

Given two positions and the time between them, the program may be used to compute elliptical orbits. This is done as follows:

1. Enter the code name for the desired body in CENTRAL BODY.

2. Enter the gravitational constant of this body for μ.

3. Set Flag 2 = 3, and input the desired positions as $\underline{r}_1$ and $\underline{r}_2$.

4. Choose values for T_1 and T_2 such that $T_2 - T_1$ is equal to the desired flight time.

6.2.5 Obtaining the Orbit Plane of Maximum Inclination for a Planet Other than Earth

If, in cases where the departure planet is not EARTH, it is desired to utilize that orbit plane having maximum inclination, Flag 3 is left as zero, the departure planet is called EARTH, the gravitational constant for the desired planet is entered as GME, and the ephemeris for this planet is entered using Flag 2 with a value of 1. If, in the same situation, $\underline{V}_{hL}$ is known, $\underline{S}_1$ may be computed from the equations

$$S_{1X} = \frac{V_{hLY}}{|\underline{V}_{hL}|}$$

$$S_{1Y} = - \frac{V_{hLX}}{|\underline{V}_{hL}|}$$

$$S_{1Z} = 0$$

Flag 3 is then set $\neq 0$, and the computed components of $\underline{S}_1$ are entered.

6.2.6 Conic Matching

The problem of matching the planetocentric hyperbolas to heliocentric ellipses in position and time may, in general terms, be solved as follows:

1. Set up the initial mission simulation on Program 291.1. The input will consist of the desired planets, closest-approach dates, closest-approach distances, and other data that may be necessary for specification of the mission. (See Applications described previously in this section.)

2. Using the results from step 1 as input, Program 281 is used to compute the planetocentric positions and velocities at the pericenters and spheres of influence of the planets of concern.

3. The Julian dates of the interaction of the planetocentric trajectories with the corresponding spheres of influence are obtained from the corresponding closest-approach dates by adding (in the case of planetary departure) or subtracting (in the case of planetary approach) the corresponding planetocentric flight times expressed in days. These flight times are obtained from Program 281 as used in step 2.

4. The dates of sphere-of-influence passage, together with the names of the respective planets, are entered into Program 291.1 to obtain the ephemerides of the desired planets at these dates. Flag 1 $\neq$ 0 for this operation.

5. The heliocentric positions of the vehicle at the spheres-of-influence are obtained by adding the corresponding planetocentric positions of the vehicle (obtained from step 2) to the heliocentric positions of the planets (obtained from step 4) at the appropriate dates.

6. Again, using Program 291.1, Flag 2 is set at 3, and the heliocentric positions of the vehicle, and the corresponding heliocentric velocities of the planets, are entered into the $\underline{r}_1$, $\underline{r}_2$, $\dot{\underline{r}}_{p1}$, $\dot{\underline{r}}_{p2}$ inputs respectively. Using a close tolerance for ϵ_T (say one second), the program is used to compute the heliocentric velocities ($\dot{\underline{r}}_1$, $\dot{\underline{r}}_2$) of the vehicle on the matched transfer ellipses.

7.0 SAMPLE CASE

Shown on the following pages is a sample case in which one leg of a three-planet flyby mission is computed. This sample consists of input data (written on the load sheet as shown) and reduced photographs of the resulting computer print.

Heliocentric and Planetocentric Orbit Determination
291.1 Fortran Version

Date 2/11/66
Page 1 of 2

1 2 3 4 5	7 8 9	11 — 72
		Engineer / Phone / Work Order / Date
2 56	BCI	, 291, WATERS , 314 , 930366 , 2/11/66
2 66	BCI	, TEST CASE, 3 PLANET FLYBY, EARTH TØ VENUS, FIRST LEG
		Planet 1
2 76	BCI	2, EARTH
		Planet 2
2 78	BCI	2, VENUS
		Time 1 (Julian Days) Fractional Days / Time 2 (Julian Days) Fractional Days
1 7	DEC	2440835. , .5 , 2440910. , .5
		Pericenter Distance at Planet 1 in km (r_p) Initialized to 6563.0
4 28	DEC	
		Flag 1 (Option for Ephemeris Data Output Only; Zero or Non-zero)
3 6	DEC	
		Flag 2 (Option for Ephemeris Data Input; Use 1 for Planet 1, 2 for Planet 2, 3 for both Planets)
3 7	DEC	
		X_1 , Y_1 , Z_1 — r_1 input in km. used if Flag 2 = 1 or 3
2 44	DEC	
		$\dot{X}_1$, $\dot{Y}_1$, $\dot{Z}_1$ — velocity of Planet 1 in km/sec used if Flag 2 = 1 or 3
2 47	DEC	
		X_2 , Y_2 , Z_2 — r_2 input in km. Flag 2 = 2 or 3
2 50	DEC	
		$\dot{X}_2$, $\dot{Y}_2$, $\dot{Z}_2$ — velocity of Planet 2 in km/sec
2 53	DEC	

Heliocentric and Planetocentric Orbit Determination
291.1 Fortran Version

Date 2/11/66
Page 2 of 2

1 2 3 4 5	7 8 9	11 — 72
		Flag 3
3 8	DEC	
		S_{1X} S_{1Y} S_{1Z}
2 41	DEC	, ,
		μ gravitational constant of central body in km^3/sec^2 (initialized as 1.3271544E+11)
4 31	DEC	
		ϵ_T tolerance in secs. on flight time computation (initialized as 5000.)
4 32	DEC	
		φ declination of Earth re-entry perifocus vector in dec. degrees (initialized as 28.5)
4 27	DEC	
		γ angle between position and velocity at Earth re-entry in dec. degrees (initialized as 96.0)
4 26	DEC	
		r_r radius of re-entry sphere at Earth (initialized as 6500.)
4 25	DEC	
		Central Body (initialized as SUN)
2 80	BCI	2,
		No. of circuits, N Approx. value of semi-major axes
6 1	DEC	,
	END	
**	FIN	

** Punch only one 'FIN'. It goes after last 'END'

291,WATERS,314,930366,2/11/66

TEST CASE, 3 PLANET FLYBY, EARTH TO VENUS, FIRST LEG

PLANET 1	PLANET 2	T*1	T*1	T*2	T*2
EARTH	VENUS	0.24408350E 07	0.50000000E 00	0.24409100E 07	0.50000000E 00
X*P1	Y*P1	Z*P1	XDOT*P1	YDOT*P1	ZDOT*P1
0.14398078E 09	-0.41124735E 08	-0.17833283E 08	0.83605300E 01	0.26007083E 02	0.11277868E 02
X*P2	Y*P2	Z*P2	XDOT*P2	YDOT*P2	ZDOT*P2
0.49454874E 06	0.88595974E 08	0.36801014E 08	-0.31243527E 02	0.13774004E 02	0.81803548E 01
XDOT*1	YDOT*1	ZDOT*1	XDOT*2	YDOT*2	ZDOT*2
0.59146712E 01	0.22501977E 02	0.93767698E 01	-0.35782735E 02	0.63266821E 01	0.28048143E 01
V*HLX	V*HLY	V*HLZ	V*HPX	V*HPY	V*HPZ
-0.24458588E 01	-0.35051060E 01	-0.19010981E 01	-0.45392079E 01	-0.74473221E 01	-0.53755406E 01
V*HL	THETA*L	PHI*L	V*HP	THETA*P	PHI*P
0.46778378E 01	0.23509271E 03	-0.23979221E 02	0.10245167E 02	0.23863733E 03	-0.31647401E 02
P	E	A	OPTION	DELTA V	INUMBER
0.10737866E 09	0.29149257E 00	0.11734961E 09	1	0.80019613E 02	6
W*X	W*Y	W*Z	T*CM	K*O	K*N
0.41501559E-02	-0.38557442E 00	0.92266741E 00	0.91416669E 07	0.10000000E 01	0.10000000E 01
A*O	EP*T	MU	CENT BODY	FLAG1	FLAG2
0.20000000E 00	0.50000000E 04	0.13271544E 12	SUN	0	0
FLAG3	X*1	Y*1	Z*1	XDOT*1	YDOT*1
0	0.00000000E-38	0.00000000E-38	0.00000000E-38	0.00000000E-38	0.00000000E-38
ZDOT*1	X*2	Y*2	Z*2	XDOT*2	YDOT*2
0.00000000E-38	0.00000000E-38	0.00000000E-38	0.00000000E-38	0.00000000E-38	0.00000000E-38
ZDOT*2	V*HPXP	V*HPYP	V*HPZP	EN	AIN
0.00000000E-38	-0.45392079E 01	-0.74473221E 01	-0.53755406E 01	0.00000000E-38	0.00000000E-38

DANGER, NEW PERICENTER LIES AHEAD

PLANETOCENTRIC

GRAV PLANET 1	GRAV PLANET 2	GRAV CNTRL BODY	R*P		
0.39860320E 06	0.32476950E 06	0.13271545E 12	0.65630000E 04		
S*1X	S*1Y	S*1Z	W*X	W*Y	W*Z
-0.82007904E 00	0.57225025E 00	0.00000000E-38	-0.23256554E 00	-0.33328448E 00	0.91369291E 00
B	B*X	B*Y	B*Z	T*X	T*Y
0.16798059E 05	0.87830496E 04	0.12586792E 05	0.68268201E 04	0.57225025E 00	0.82007905E 00
T*Z	R*X	R*Y	R*Z	B.T	B.R
0.00000000E-38	0.00000000E-38	0.00000000E-38	-0.99999998E 00	0.15348267E 05	-0.68263199E 04
A	E	A*1	A*2	A*3	A*4
-0.18215893E 05	0.13602898E 01	0.87835386E-03	-0.00000000E-38	-0.66177229E-12	-0.90823476E-17

P*X	P*Y	P*Z	K	R*OPT
-0.24841394E 00	0.92864686E 00	0.27550948E 00	0.90000004E 02	0.75452692E 04

OUTGOING HYPERBOLA

P'*X	P'*Y	P'*Z	R'*P	E'	A'
-0.17104528E 00	0.93852543E 00	0.29891520E 00	0.65504496E 04	0.13596008E 01	-0.18215893E 05

OMEGA
0.46692747E 01

REFERENCES

[1] Knip, G. K., (Jr.), Three-Dimensional Sphere-of-Influence Analysis of Interplanetary Trajectories to Mars, NASA Technical Note D-1199

[2] Peabody, P. R., et al, User's Description of JPL Ephemeris Tapes, TR 32-580, March 2, 1964

[3] Jordon, James F., The Application of Lambert's Theorem to the Solution of Interplanetary Transfer Problems, TR 32-521, February 1, 1965

PART 1-2

PROGRAM DESCRIPTION

FOR

STATE DETERMINATION OF

HYPERBOLIC ORBITS

ABSTRACT

This section contains the description of a digital computer program for the determination of position, velocity, and flight time of planetocentric hyperbolic orbits. The output of this program determines initial conditions for nominal planetocentric orbits as used in the performance assessment of space guidance systems.

TABLE OF CONTENTS

1.0 INTRODUCTION

Program Deck 281 is used in connection with Program 291.1 in order to determine the initial conditions for the nominal two-body orbits as used in the performance assessment program for the free-fall phases of interplanetary missions. The connection between the two programs was explained in the introduction of the description for Program Deck 291.1, and will, therefore, not be repeated here. The computations consist of a straightforward application of two-body formulas in order to obtain from the output of Program 291.1 the position and velocity at prespecified distances from the center of the planet. A detailed discussion is contained in Paragraph 2 of the description.

1.1 DEFINITION OF MATHEMATICAL SYMBOLS

The following symbols are used in the program to be described.

a	semi-major axis of hyperbola
A or A(T)	matrix transformation expressing mean equatorial coordinate system of 1950.0 in terms of the mean equatorial coordinate system of T_L
e	eccentricity of hyperbola
E or E(τ)	matrix transformation expressing the ecliptic coordinate system in terms of the equatorial coordinate system
E^T	the transpose of E
$\underline{I}, \underline{J}, \underline{K}$	an irrotational, right-handed, planet-centered coordinate system with $\underline{I}$ and $\underline{J}$ in the Earth's equatorial plane. $\underline{I}$ is along the vernal equinox and $\underline{K}$ along the north pole.
$\underline{i}, \underline{j}, \underline{k}$	an irrotational, right-handed, planet-centered coordinate system with $\underline{i}$ and $\underline{j}$ in the ecliptic plane and $\underline{k}$ along the north celestial pole.
p	semi-latus rectum of hyperbola
$\underline{P}$	a unit vector directed to pericenter expressed in the "mean ecliptic and equinox of time T_L" under normal program usage
q	perifocal distance
$\underline{r}$	position vector (magnitude r)

T	time interval in Julian centuries from 1950.0
T_L	Julian days of time defining coordinate system
$\underline{U}_{rp}$	incoming asymptote vector
$\underline{v}$	velocity vector
$\underline{V}_{hp}$	hyperbolic-excess velocity vector
V_{hp}	hyperbolic-excess speed, the magnitude of $\underline{V}_{hp}$
$\underline{W}$	a unit vector perpendicular to the vehicle's orbit plane oriented such that $\underline{Q} = \underline{W} \times \underline{P}$ is parallel to the direction of motion of the vehicle at pericenter
β_p	declination of $\underline{U}_{rp}$ in decimal degrees
ϵ	mean obliquity of the ecliptic
θ_{sp}	right ascension of the outgoing asymptote in decimal degrees
λ_p	longitude of $\underline{U}_{rp}$ in decimal degrees
μ	gravitational constant of planet
τ	time interval in Julian centuries since 1900.0
ϕ_{sp}	declination of the outgoing asymptote in decimal degrees
sign	(sign = ± 1) input to reverse the direction of the $\underline{W}$ vector. Normal value is +1
n	($n = \pm 1$) input to determine the quadrant for computation. Outgoing trajectories require that $n = -1$, incoming trajectories require that $n = 1$.
R_1	position 1 for flight time determination
R_2	position 2 for flight time determination
υ	mean motion
x_1, y_1, z_1	mean ecliptic coordinates of T_L
x_2, y_2, z_2	mean equatorial coordinates of T_L
x_3, y_3, z_3	mean equatorial coordinates of 1950.0
x_4, y_4, z_4	mean ecliptic coordinates of 1950.0

2.0 MATHEMATICAL MODEL

2.1 GENERAL DESCRIPTION OF THE PROBLEM

As stated earlier, the problem defined here is that of computing position and velocity on hyperbolic orbits in a inverse-square law force field. The asymptotes of the hyperbola are specified as unit vectors in spherical coordinates relative to any arbitrary Cartesian system. The two asymptotes define the plane of motion having unit normal vector $\underline{W}$, while orbital computations are performed relative to the conventional $\underline{P}$, $\underline{Q}$,$\underline{W}$ system illustrated in Figure 1. The vector $\underline{P}$ (see Figure 1) is formed from a linear combination of the asymptote vectors $\underline{V}$ and $\underline{S}_2$ (see Eq. 4 in Paragraph 5.3.2), while $\underline{Q}$ is defined by $\underline{W} \times \underline{P}$. In this manner a right-handed set of in-plane coordinates is formed.

The scalar quantities a and e are computed from the hyperbolic-excess speed V_{hp} and the closest-approach distance q from the relationships

$$a = \frac{-\mu}{V_{hp}^{2}} \tag{1}$$

and

$$e = 1 - \frac{q}{a} \tag{2}$$

The asymptote angle α and the in-plane coordinates x_ω, y_ω of the position vector $\underline{r}$ are then computed from a, e, and r, where r is the magnitude of the radius vector (an input quantity). Clearly, in order to preserve consistency in definition, $r \geq q$.

Flight time is computed from the hyperbolic anomaly, F, the mean motion, υ, and two independent input distances R_1 and R_2. As this relationship involves scalars only, it is clear that R_1 and R_2 must lie on the same side of the major axis. The basic equation is

$$t_f = \frac{(U_1 - U_2) + F_2 - F_1}{\upsilon} \tag{3}$$

where

$$\upsilon = \sqrt{\frac{\mu}{|a|^3}}$$

and

$$U_i = \sqrt{\left(\frac{|a| + R_i^2}{|a|\,e}\right) - 1} \qquad (i = 1, 2)$$

and

$$F_i = \sinh^{-1} U_i$$

t_f is the flight time expressed in units consistent with r, μ, and V_{hp}.

2.2 CHOICE OF PROPER QUADRANT

Some ambiguity will result unless the proper quadrant for $\underline{r}$ is chosen. In this program quadrant choice is made on the basis of the argument that motion with respect to a point may be divided into three general categories:

1. Approaching the point,
2. Leaving the point,
3. Neither approaching nor leaving the point.

Proper quadrant choice is made through use of the input quantity n, where n = ± 1 in accordance with the following scheme:

1. If the desired trajectory approaches the planet, n is input as 1.
2. If the desired trajectory is leaving the planet, n is input as -1.

At pericenter the vehicle is neither approaching nor leaving, and either +1 or -1 may be used. In the equations the effect of using ± 1 for n is that of reflecting the y_ω axis, and hence the geometry of motion, about the $\underline{P}$ vector. A logic test is introduced so that the proper asymptote (incoming or outgoing) is chosen.

The direction of the orbit plane vector $\underline{W}$ may be reversed by changing the sign of the $|\pm 1|$ in the "sign" input (see Eq. 11, Paragraph 5.3.1).

2.3 ADDITIONAL COORDINATE SYSTEMS

If $\underline{U}_{rp}$, the incoming asymptote vector, is expressed relative to the mean ecliptic system of date T_L (T_L is an input time expressed in Julian days), and ϕ_{sp}, λ_{sp} (spherical components of the outgoing asymptote $\underline{S}_{ei}$) are expressed relative to the mean equatorial system of 1950.0, then position and velocity will be computed in both the ecliptic and equatorial systems. The mean obliquity of the ecliptic ϵ is computed from

$$\epsilon = 23^o.452294 - (0^o.0130125)\,\tau - (0^o.164 \times 10^{-5})\,\tau^2 \;*$$

* Space Trajectories Program for the IBM 7090 Computer, by D. B. Holdridge; JPL Technical Report No. 32-223

where

$$\tau = \frac{T_L - 2415020.3}{36525.}$$

and T_L (an input) is the Julian date at which the transformation is to take place. A single rotation about the axis of the 1950.0 vernal equinox is performed from the rotation matrix

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \epsilon & \sin \epsilon \\ 0 & -\sin \epsilon & \cos \epsilon \end{bmatrix}$$

This transforms the outgoing asymptote defined by ϕ_{sp}, λ_{sp} from the mean equatorial system of date T_L to the mean ecliptic and equinox system of T_L.

However, should both asymptotes be entered in terms of a single coordinate system, the value 57697945. should be used for T_L. This will compute ϵ to be 0^o.

Transformation to the ecliptic and equatorial systems of 1950.0 is accomplished by the A(T) rotation matrix defined by equations 15, 16, 17 in Paragraph 5.3.2.* However, this transformation becomes meaningless if T_L = 57697945.

3.0 ORGANIZATION OF THE PROGRAM

Flow charts provide the basic framework around which the discussion is constructed. These diagrams serve to indicate the logical flow connecting different functional blocks. They do not describe literally the operation within the computer program itself because many of the programming details are of little interest to most engineers.

The flow charts have been arranged and drawn according to a heirarchical structure. The "highest" level, designated as Level I, depicts the overall structure of the program. Each block appearing in this chart is described by another flow chart. These charts are designated as Level II. This policy is repeated for each block in every level until no further logic remains to be described.

Paragraph 3.1 contains a further discussion and definition of the criteria used to establish the different flow chart levels. The symbols used in the flow charts are defined in Paragraph 3.2. The symbols and nomenclature that are fundamental to the discussion and equations are defined in Paragraph 3.3.

---

* Space Trajectories Program for the IBM 7090 Computer, by D. B. Holdridge; JPL Technical Report No. 32-223

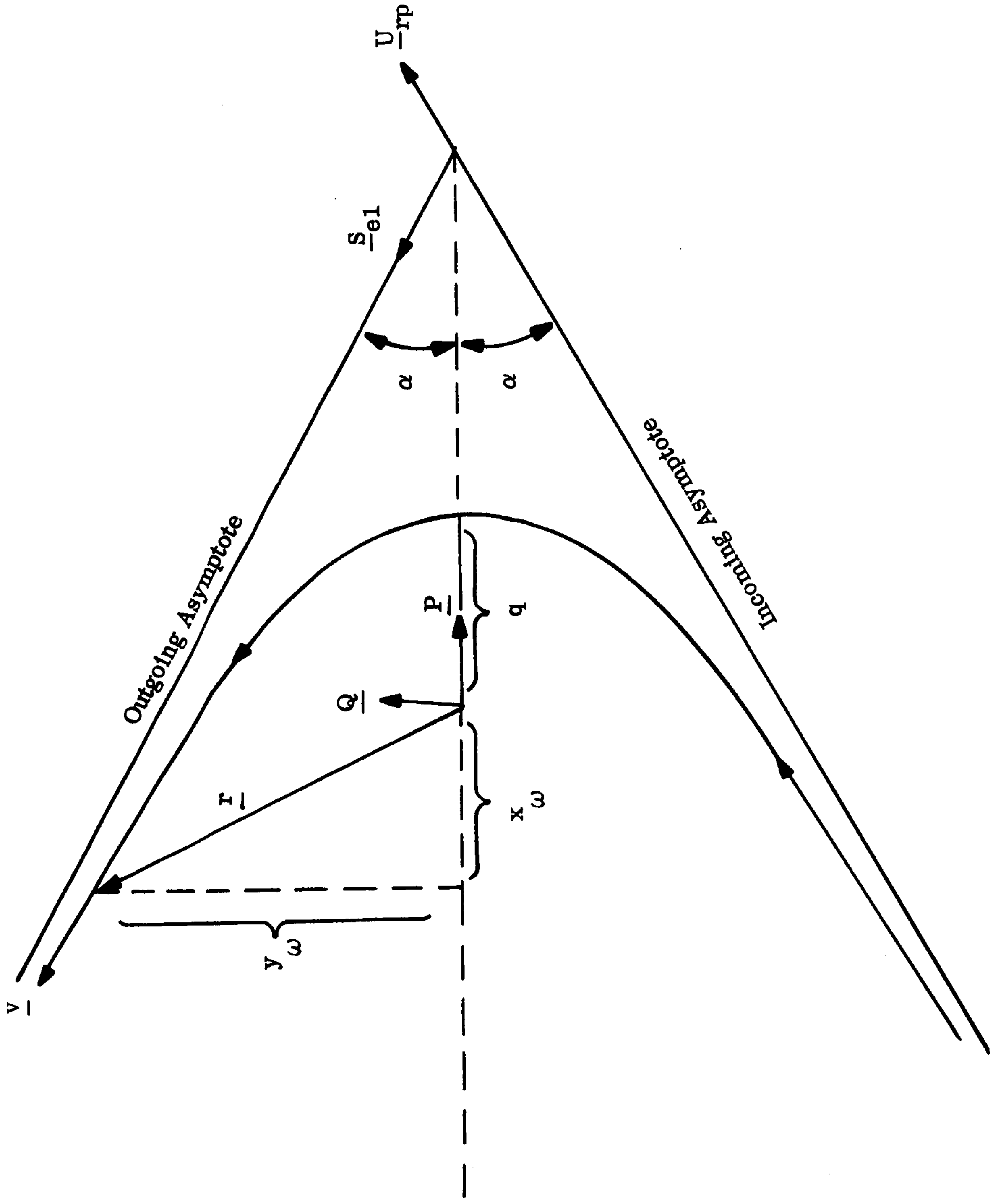


Figure 1. Geometry of Hyperbola

3.1 SCHEMA FOR FLOW CHART PRESENTATION

As has already been stated, the flow charts are arranged according to "levels". In the resulting heirarcy, the Level I flow chart provides the most general description since it depicts the overall program. Each functional block is further described by lower level flow charts. These charts indicate the logical flow within the block and describe the input and output requirements of the block. The equations used to obtain the desired outputs are presented as a supplement to the lowest level flow chart.

LEVEL I: This flow chart is designed to provide a very general description of the entire program. The titles assigned to the functional blocks are intended to be suggestive of the nature of the role to be performed within the block. Those functions that are to be performed in the basic computational cycle are designated by Roman numerals. Arabic symbols are used for functions that occur only once or play a passive role.

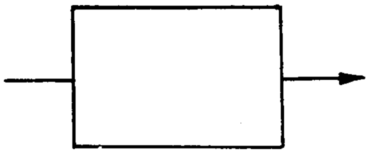
To indicate the basic logical decisions that can regulate and alter the flow between functional blocks, decision blocks are indicated. These decisions represent in a general manner the types of decisions that are required. The actual decision logic is described in the Level II flow charts of the functional blocks immediately preceding the decision block.

LEVEL II: The Level II flow charts provide the first concrete description of the program. Only the most important logical flow within each functional block is indicated on these diagrams. The quantities that are required for all logical and computational operations within this block are stated on this chart. These quantities are differentiated as being either INPUT (i.e., values provided initially by the engineer) or COMPUTED (i.e., values determined in other portions of the program). The quantities that are required in other parts of the program, either for print-out or for computations, are also indicated on this flow chart. The functional blocks that appear on these diagrams are denoted by two symbols (e.g., II.1 when discussing the "first" block in the Level II flow chart of functional block II) and a name. The names have been selected to provide some insight into the nature of the block.

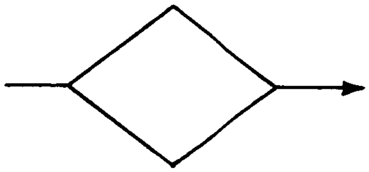
LEVEL III: These diagrams provide additional details of the logic flow within the functional blocks depicted at Level II. In this program definition, Level III provides the description of the most intimate logical details in almost every case so no purpose was served by proceeding to lower levels. These flow diagrams are augmented by the equations programmed into the computer. The input and output requirements of these blocks are stated on the diagrams. All of these quantities are summarized on the Level II flow chart.

3.2 DEFINITION OF FLOW CHART SYMBOLS

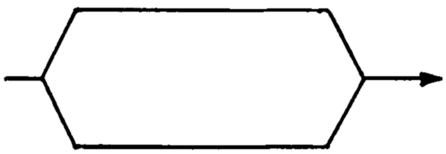
The following symbols represent the only ones that are used in the flow charts presented below.



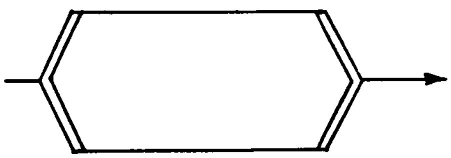
Set of operations that is to be described further by additional flow charts or by equations



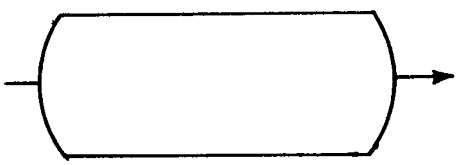
Logical Decision



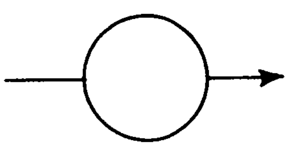
Operations that are predefined (i.e., in some other document)



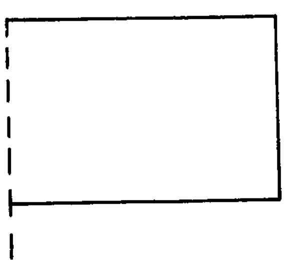
Operations are completely defined by the statements contained within the box



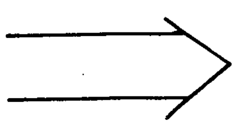
Connector used on Level II flow charts to indicate entry source and exit destination



Connector used on Level III flow charts



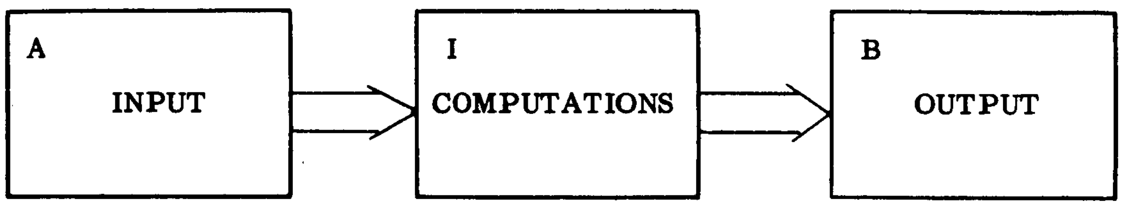
Summary of all quantities required in computations of flow chart on which this symbol appears or, alternatively, summary of all quantities computed in this flow chart which are required in other operations.



This broad arrow appears on Level I and Level II flow charts. It is used to indicate information flow from one block to another. The more important information is stated within the arrow. This symbol has been introduced to emphasize that many quantities are transmitted between the functional blocks in the higher level charts.

3.3 BASIC ORGANIZATION OF THE PROGRAM

In general, the program consists of a straightforward sequence of computations using a certain input format. This overall scheme is illustrated in the Level I flow chart below.



Level I Flow Chart

4.0 <u>INPUT, OUTPUT</u>

4.1 INPUT - BLOCK A

4.1.1 <u>Input Quantities</u>

The program input consists of heading or identification data plus the following quantities.

V_{hp}	hyperbolic excess speed (units of length/time)
q	closest approach distance (units of length)
T_L	(time in Julian days; if the asymptote vectors are input relative to the same coordinate system, then T_L must be input as 57697945.)
μ	gravitational constant of central body (length^3/time^2)
r	magnitude of position vector in units of length
β_p	declination of the incoming asymptote in decimal degrees
λ_p	right ascension of the incoming asymptote in decimal degrees
ϕ_{sp}	declination of the outgoing asymptote in decimal degrees
θ_{sp}	right ascension of the outgoing asymptote in decimal degrees
sign	may determine direct or retrograde motion; equal to ± 1; normally equal to +1

n	quadrant determination factor equal to ± 1; n = -1 for outgoing trajectory, +1 for incoming trajectory
R_1	position 1 for flight time determination, in units of length
R_2	position 2 for flight time determination, in units of length
Note:	The program will compute the time of flight from distance R_1 to distance R_2. However, R_1 and R_2 must be on the same side of the major axis

4.1.2 Load Sheet

The above input quantities are arranged on the standardized load sheet shown on the following page.

4.2 OUTPUT - BLOCK C

4.2.1 Output Quantities

The program prints all of the major quantities computed in the format shown in the check case illustrated in Paragraph 6.3. These are defined below in terms of the notation symbols used in the computer output. See Paragraph 1.3 for complete definitions.

V*HP	V_{hp}	X*O	components of $\underline{r}$
Q	q	Y*O	
T*L	T_L	Z*O	
MU	μ	X.*O	components of $\underline{v}$
R*O	r	Y.*O	
B*P	β_p	Z.*O	
LAMBDA*P	λ_p	Note: $\underline{r}$ and $\underline{v}$ are printed in four co-ordinate systems, each labeled as shown on the sample output form.	
PHI*SP	ϕ_{sp}		
THETA*SP	θ_{sp}	P*X	components of $\underline{P}$
SIGN	sign (± 1)	P*Y	
N	n	P*Z	

PROGRAM 281

INITIAL STATE DETERMINATION

Date ________
Page _1_ of _1_

1 2 3 4 5	7 8 9	11
		Engineer / Phone / Work Order / Date
1 1	BCI	, 281.1 , , , ,
		Heading (56 characters)
1 11	BCI	,
		V_{hp} (hyperbolic excess speed in units of length/time) q(closest approach distance in units of length)
1 21	DEC	,
		T_L (time in Julian days) μ (central body's gravitational constant in units of $length^3/time^2$)
1 23	DEC	,
		r (magnitude of position vector in units of length) β_p (declination of the incoming asymptote in units of decimal degrees)
1 25	DEC	,
		λ_p (right ascension of the incoming asymptote in units of decimal degrees)
1 27	DEC	
		φ_{sp} (declination of the outgoing asymptote in units of decimal degrees)
1 28	DEC	
		θ_{sp} (right ascension of the outgoing asymptote in units of decimal degrees)
1 29	DEC	
		Sign (multiplying factor to define positive or retrograde motion)
1 30	DEC	
		n (-1. for outgoing; +1 for incoming trajectory)
1 31	DEC	
		R_1 (position 1 for flight time determination in units of length)
1 32	DEC	
		R_2 (position 2 for flight time determination in units of length)
1 33	DEC	
	END	

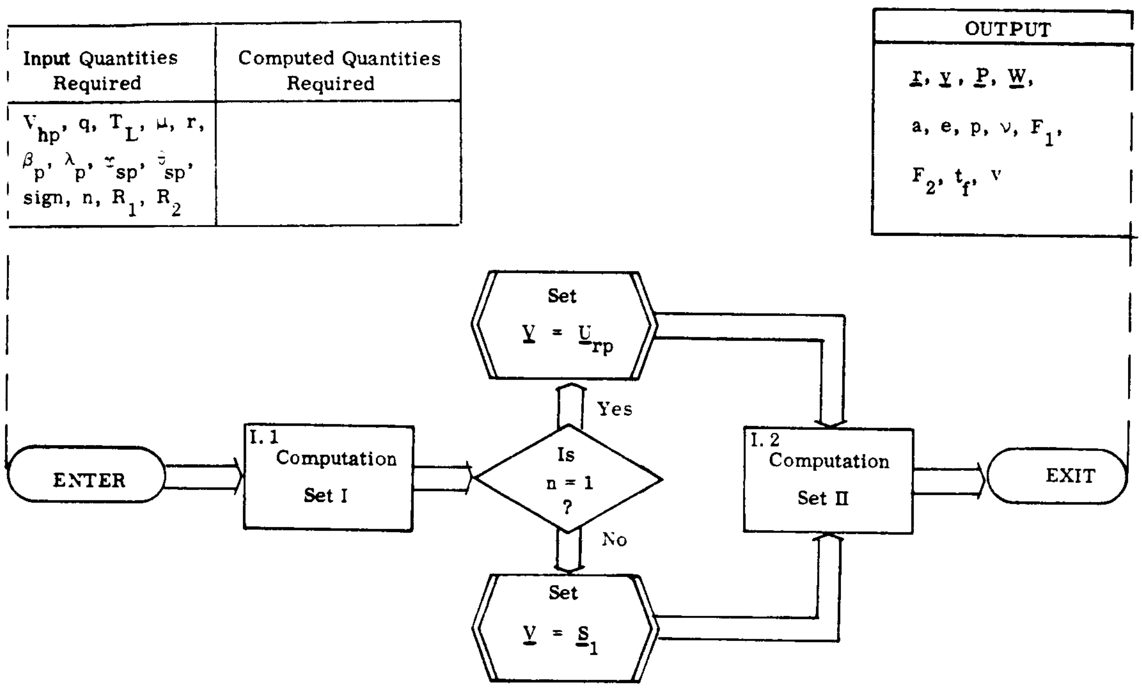
A "FIN" must follow "END" of last case

R*1	R_1		QI	Q_x	components of $\underline{Q}$
R*2	R_2		QJ	Q_y	
NU	υ		QK	Q_z	
F*1	F_1		URPI	U_{rpx}	components of $\underline{U}_{rp}$
F*2	F_2		URPJ	U_{rpy}	
T*F	t_f		URPK	U_{rpz}	
V*O1	$\lvert\underline{v}\rvert$		WI	W_x	components of $\underline{W}$
A	a		WJ	W_y	
E	e		WK	W_z	
P	p		S2I	S_{2x}	components of $\underline{S}_2$
VC	v_c		S2J	S_{2y}	
SIN ALPHAC	$\sin \alpha_c$		S2K	S_{2z}	
COS ALPHAC	$\cos \alpha_c$		SIN ALPHA	$\sin \alpha$	
SXE	S_{xe}	components of $\underline{S}_e$	COS ALPHA	$\cos \alpha$	
SYE	S_{ye}		V*1	v_1	
SZE	S_{ze}		NU	ν	
TAU	τ		F1	F_1	
XW	x_ω		F2	F_2	
YW	y_ω		TF	t_f	
SX1	S_{x1}	components of $\underline{S}_1$	U1	U_1	
SY1	S_{y1}		U2	U_2	
SZ1	S_{z1}		A11,...,A33	$a_{11},\ldots,a_{33}$	

5.0 COMPUTATION BLOCK I

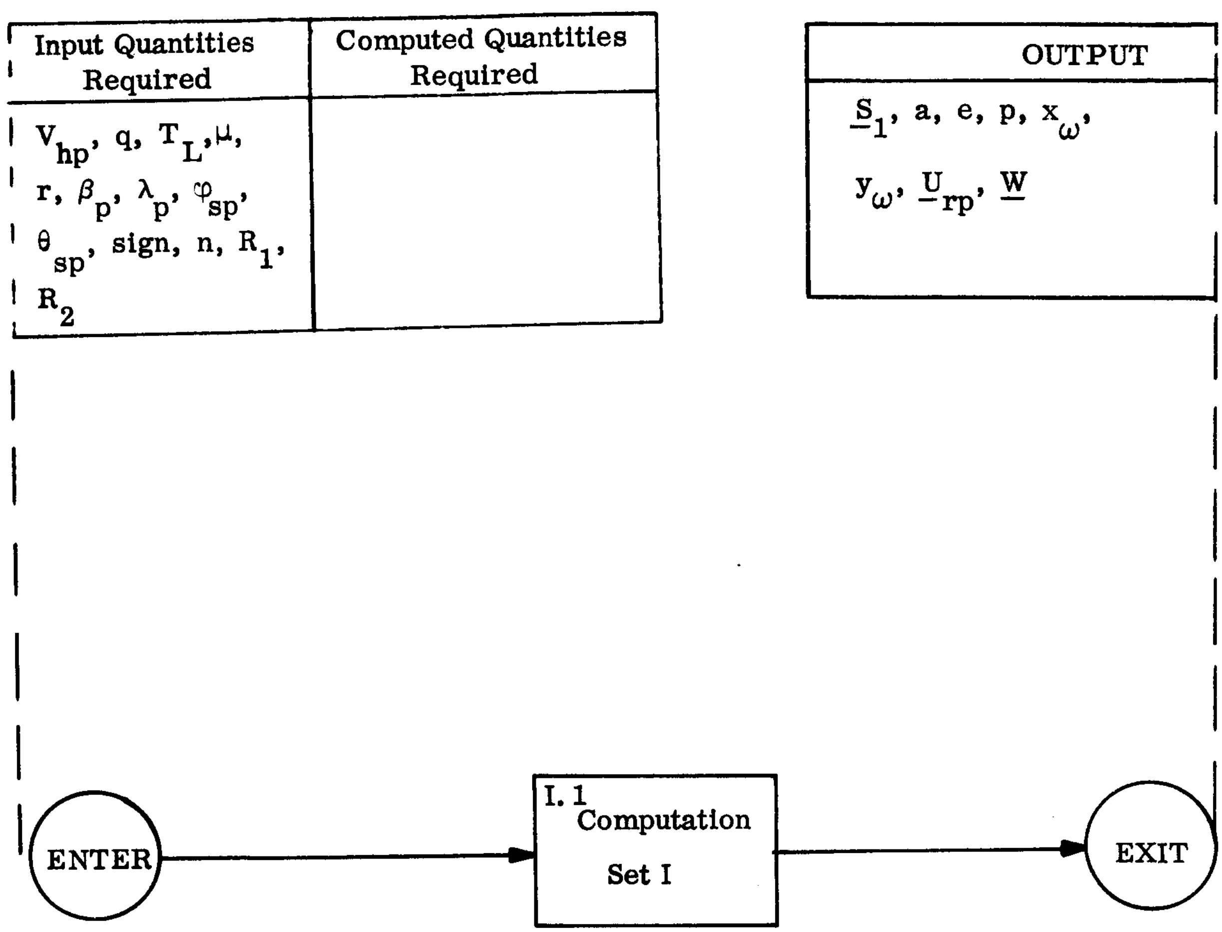
5.1 LEVEL II FLOW CHART

The computation block is defined more specifically by the Level II flow chart shown below.



5.2 LEVEL III FLOW CHARTS

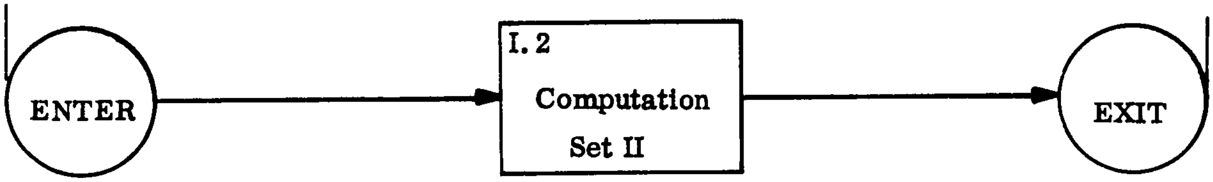
5.2.1 Computation Set I



5.2.2 Computation Set II

Input Quantities Required	Computed Quantities Required
	$\underline{S}_1$, a, e, p, x_ω, y_ω, $\underline{U}_{rp}$, $\underline{W}$

OUTPUT
$\underline{r}$, $\underline{v}$, $\underline{P}$, $\underline{W}$, a, e, p, ν, F_1, F_2, t_f, v



5.3 DETAILED EQUATIONS

5.3.1 Computation Set I

1.
$$\underline{S}_{e1} = \cos\phi_{sp} \cos\theta_{sp} \underline{I} + \cos\phi_{sp} \sin\theta_{sp} \underline{J} + \sin\phi_{sp} \underline{K}$$

$$\triangleq S_{xe} \underline{I} + S_{ye} \underline{J} + S_{ze} \underline{K}$$

2.
$$\tau = \frac{T_L - 2415020.3}{36525.}$$

3.
$$\epsilon = 23^{o}.452294 - (0^{o}.0130125)\,\tau - (0^{o}.164 \times 10^{-5})\,\tau^2$$

4.
$$\underline{S}_1 = \begin{bmatrix} S_{x1} \\ S_{y1} \\ S_{z1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\epsilon & \sin\epsilon \\ 0 & -\sin\epsilon & \cos\epsilon \end{bmatrix} \begin{bmatrix} S_{xe} \\ S_{ye} \\ S_{ze} \end{bmatrix} \triangleq E \begin{bmatrix} S_{xe} \\ S_{ye} \\ S_{ze} \end{bmatrix}$$

5.
$$a = \frac{-\mu}{V_{hp}^2}$$

6.
$$e = 1 - \frac{q}{a}$$

7.
$$p = a(1 - e^2)$$

8.
$$x_\omega = \frac{1}{e}(p - r)$$

9.
$$y_\omega = -n\sqrt{r^2 - x_\omega^2}$$

10.
$$\underline{U}_{rp} = \cos\beta_p \cos\lambda_p \underline{i} + \cos\beta_p \sin\lambda_p \underline{j} + \sin\beta_p \underline{k}$$

11.

$$\underline{W} = \frac{\underline{U}_{rp} \times \underline{S}_1}{|\underline{U}_{rp} \times \underline{S}_1|} \quad \text{(sign)}$$

5.3.2 Computation Set II

1.

$$\underline{S}_2 = \underline{V} \times \underline{W} \quad \text{where}$$

$$\underline{V} = \begin{cases} \underline{U}_{rp} & \text{if } n = 1 \\ \underline{S}_1 & \text{if } n = -1 \end{cases}$$

2.

$$\sin \alpha = \sqrt{\frac{e^2 - 1}{e^2}}$$

3.

$$\cos \alpha = \frac{n}{e}$$

4.

$$\underline{P} = \underline{V} \cos \alpha + \underline{S}_2 \sin \alpha$$

5.

$$\underline{Q} = \underline{W} \times \underline{P}$$

6.

$$\underline{r}_1 = x_\omega \underline{P} + y_\omega \underline{Q}$$

7.

$$v_c = \sqrt{V_{hp}^2 + \frac{2\mu}{r}}$$

8.

$$\sin \alpha_c = \frac{|(x_\omega + a\,e)(e^2 - 1)|}{\sqrt{(x_\omega + a\,e)^2 (e^2 - 1)^2 + y_\omega^2}}$$

9.

$$\cos \alpha_c = \frac{n\,|y_\omega|}{\sqrt{(x_\omega^2 + a\,e)^2 (e^2 - 1)^2 + y_\omega^2}}$$

10.

$$\underline{v}_1 = \underline{P}\,v_c \cos \alpha_c + \underline{Q}\,v_c \sin \alpha_c$$

11.
$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = E^T \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = E^T \underline{r}_1$$

12.
$$\underline{v}_2 = E^T \underline{v}_1$$

13.
$$t_f = \frac{e\,(U_1 - U_2) + F_2 - F_1}{\upsilon}$$

where
$$\upsilon = \sqrt{\frac{\mu}{|a|^3}} \; ; \quad U_i = \sqrt{\left(\frac{|a| + r_i}{|a|\, e}\right)^2 - 1} \qquad i = 1, 2$$

$$F_i = \sinh^{-1} U_i$$

14.
$$v_1 = (\dot{x}_1^{\,2} + \dot{y}_1^{\,2} + \dot{z}_1^{\,2})^{1/2}$$

15.
$$T = \frac{T_L - 2433282.4}{36525.}$$

16.
$$a_{11} = 1 - (.29697E-3)\, T^2 - (.13E-6)\, T^3$$
$$a_{12} = -a_{21} = -(.2234988E-1)\, T - (.676E-5)\, T^2 + (.221E-5)\, T^3$$
$$a_{13} = -a_{31} = -(.971711E-2)\, T + (.207E-5)\, T^2 + (.96E-6)\, T^3$$
$$a_{22} = 1 - (.24976E-3)\, T^2 - (.15E-6)\, T^3$$
$$a_{23} = a_{32} = -(.10859E-3)\, T^2 - (.3E-7)\, T^3$$
$$a_{33} = 1 - (.4721E-4)\, T^2 + (.2E-7)\, T^3$$

17.

$$A \triangleq \begin{bmatrix} a_{11} & -a_{12} & -a_{13} \\ -a_{21} & a_{22} & a_{23} \\ -a_{31} & a_{32} & a_{33} \end{bmatrix}$$

18.

$$\underline{r}_3 = A \, \underline{r}_2$$

19.

$$\underline{v}_3 = A \, \underline{v}_2$$

20.

$$\epsilon = 23.445787$$

21.

$$\underline{r}_4 = E \, \underline{r}_3$$

22.

$$\underline{v}_4 = E \, \underline{v}_3$$

6.0 USER'S GUIDE

6.1 USE OF INPUT

The input format to this program is shown in Paragraph 4.2.1. Although the form simple and straightforward, there are some points of concern which will be repeated here for emphasis.

6.1.1 Care must be taken to insure that all units are correct and consistent with each other. The date T_L must be expressed in Julian days, and the incoming and outgoing asymptotes must not be interchanged.

6.1.2 The positions R_1, R_2 for flight time determination must lie on the same side of the axis of symmetry of the hyperbola. If $R_2 > R_1$, the computed flight time t_f will appear as a negative number. Its absolute value will, however, be the same if R_1 and R_2 are interchanged.

6.1.3 Care must be taken to ascertain the correct state of motion, i.e., whether the vehicle is approaching or leaving the central body. This in turn is reflected in the proper use of n, defined as follows:

1. If the desired trajectory approaches the planet, let n = 1.
2. If the desired trajectory is leaving the planet, let n = -1.
3. If neither of the above is the case (evaluation at pericenter), either +1. or -1. may be used.

PARTICULAR ATTENTION MUST BE GIVEN TO THIS POINT IF THE DESIRED RESULTS ARE TO BE OBTAINED.

6.1.4 Although the orbit plane and direction of motion (whether direct or retrograde) are originally defined by the asymptote vectors, another vector in the plane may be used in place of one asymptote vector, provided that:

1. If the vehicle is approaching the central body, the incoming asymptote remains as defined.
2. If the vehicle is leaving the central body, the outgoing asymptote remains as defined.
3. The cross product of the vectors used thusly results in a vector having the same direction as the cross product of incoming and outgoing asymptotes, in that order (see Eq. 11 in Paragraph 5.3.1). Should the resulting $\underline{W}$ vector be opposite in direction to its desired orientation, - 1 should be used in the "sign" input.

6.1.5 If ϕ_{sp}, θ_{sp}, β_p, λ_p are expressed relative to the same coordinate system, the value 57697945. must be used for T_L. In this, the usual case, the printout corresponding to the 1950.0 system (see output of check case) will be meaningless. However, the first two coordinate systems printed should give identical components.

6.2 SAMPLE RUN - OUTPUT FORMAT

A check case containing computer print is shown on the following pages.

PROGRAM 281

INITIAL STATE DETERMINATION

Date 1-19-66
Page 1 of 1

1 2 3 4 5	7 8 9	11
1 1	BCI	, 281.1 , WATERS , 314 , 930366 , 1-19-66
		Heading (56 characters)
1 11	BCI	,CHECK ① CASE ② MARTIAN ① FLYBY ② L-5 ② LEAVE ① EARTH ① AT ① INJ.
		V_{hp} (hyperbolic excess speed in units of length/time) q(closest approach distance in units of length)
1 21	DEC	.514335 87 E1 , .6563 E4
		T_L (time in Julian days) μ (central body's gravitational constant in units of length^3/time^2)
1 23	DEC	.57697945 E8 , .39860320 E6
		r (magnitude of position vector in units of length) β_p (declination of the incoming asymptote in units of decimal degrees)
1 25	DEC	.6563 E4 , 0
		λ_p (right ascension of the incoming asymptote in units of decimal degrees)
1 27	DEC	.20892632 E3
		φ_{sp} (declination of the outgoing asymptote in units of decimal degrees)
1 28	DEC	-.30813455 E2
		θ_{sp} (right ascension of the outgoing asymptote in units of decimal degrees)
1 29	DEC	.29892632 E3
		Sign (multiplying factor to define positive or retrograde motion)
1 30	DEC	1.
		n (-1. for outgoing; +1 for incoming trajectory)
1 31	DEC	-1.
		R_1 (position 1 for flight time determination in units of length)
1 32	DEC	.6563 E4
		R_2 (position 2 for flight time determination in units of length)
1 33	DEC	.924 E6
	END	

A "FIN" must follow "END" of last case

```
                    281.1.WATERS.314.930366.1-19-66

                    CHECK CASE  MARTIAN FLYBY  L-S  LEAVE EARTH AT INJ.

INITIAL STATE DETERMINATION FOR HYPERBOLIC ORBITS

   V*HP              Q                 T*L               MU                R*0               B*P
 0.51433586E 01    0.65630000E 04    0.57697945E 08    0.39860320E 06    0.65630000E 04    0.

   LAMBDA*P          PHI*SP            THETA*SP          SIGN              N                 R*1
 0.20892632E 03   -0.30813455E 02    0.29892632E 03    0.10000000E 01   -0.10000000E 01    0.65630000E 04

   R*2
 0.92400000E 06

   X*0               Y*0               Z*0               X.*0              Y.*0              Z.*0
MEAN ECLIPTIC COORDINATE SYSTEM OF LAUNCH DATE
-0.60204293E 04    0.11589693E 04    0.23418340E 04   -0.37903069E 01   -0.10657262E 02   -0.44699292E 01

MEAN EQUATORIAL COORDINATE SYSTEM OF LAUNCH DATE
-0.60204293E 04    0.11589690E 04    0.23418341E 04   -0.37903069E 01   -0.10657261E 02   -0.44699304E 01

MEAN EQUATORIAL COORDINATE SYSTEM OF 1950.0
-0.97811707E 07   -0.47880899E 08   -0.20457399E 08    0.10015927E 06   -0.15632980E 05   -0.86406245E 04

MEAN ECLIPTIC COORDINATE SYSTEM OF 1950.0
-0.97811707E 07   -0.52067319E 08    0.28253375E 06    0.10015927E 06   -0.17780216E 05   -0.17071595E 04

   P*X               P*Y               P*Z               W*X               W*Y               W*Z
-0.91732886E 00    0.17659144E 00    0.35682373E 00    0.24776470E 00   -0.44833811E 00    0.85883968E 00

   A                 E                 P                 VC                SIN ALPHAC        COS ALPHAC
-0.15067707E 05    0.14355672E 01    0.15984628E 05    0.12162399E 02    0.10000000E 01   -0.23179961E-07

   SXE               SYE               SZE               TAU               XW                YW
 0.41540740E 00   -0.75169288E 00   -0.51224456E 00    0.15135639E 04    0.65629999E 04    0.37052303E-03

   SX1               SY1               SZ1               QI                QJ                QK
 0.41540740E 00   -0.75169294E 00   -0.51224447E 00   -0.31164141E 00   -0.87624675E 00   -0.36752035E 00

   URPI              URPJ              URPK              WI                WJ                WK
-0.87524244E 00   -0.48368447E 00    0.                0.24776470E 00   -0.44833811E 00    0.85883968E 00

   S2I               S2J               S2K               SIN ALPHA         COS ALPHA         V*1
-0.87524243E 00   -0.48368446E 00    0.55879354E-08    0.71747063E 00   -0.69658875E 00    0.12162398E 02

   NU                F1                F2                TF                J1                U2
 0.34134980E-03    0.                0.44637882E 01   -0.16945337E 06    0.                0.43402121E 02

   A11               A12               A13
-0.11291858E 04   -0.76060412E 04   -0.33154304E 04

   A21               A22               A23
 0.76060412E 04   -0.10903840E 04   -0.35252045E 03

   A31               A32               A33
 0.33154304E 04   -0.35252045E 03   -0.37801766E 02
```

PART 1-3

DESIGN OF NOMINAL INTERPLANETARY MISSIONS

ABSTRACT

This section describes a general procedure for the design of an interplanetary free-fall trajectory using Programs 281.1 and 291.1. As each of these programs has been discussed in detail in the preceding parts, only the input-output transfer and manipulation is discussed here. If a more detailed knowledge of mission design is desired, two references are listed at the end of the text.

PROCEDURE FOR DESIGNING NOMINAL INTERPLANETARY MISSIONS

1. Select the desired planets to be encountered and the corresponding dates of encounter [1]. Other information such as closest approach distances and earth re-entry data may be chosen, if desired. This information is then entered as input to program 291.1. The program is then run for computation of the approximate heliocentric and planetocentric parameters.

2. The next step is that of establishing the near-planet or planetocentric phases of the mission. These start at entry into the planetary sphere of influence [2], and end upon exit from this sphere. In turn, the planetocentric phases are divided into two additional phases, incoming and outgoing. The incoming planetocentric phase starts at entry into the sphere of influence and ends upon reaching the closest approach distance to the planet (this distance is chosen by the designer in accordance with his particular criteria.1.1 planetary radius is a good rule-of-thumb value). The outgoing planetocentric phase starts at the closest approach point to the planet, and ends at the point of exit from the sphere of influence. The time of closest approach is generally chosen to be the Julian date entered in program 291.1 for the particular planet, while the times at which the spheres of influence are crossed are obtained by adding or subtracting the corresponding flight times as obtained from program 281.1.

   To obtain the outgoing planetocentric positions, velocities, and flight times the following output data from the first 291.1 computation is entered as input data for 281.1:

Output Quantity from 291.1	Input Quantity for 281.1
V*HL	V_{hp}
R*P	q
GRAV PLANET 1	μ
R*P	r
PHI*P †	β_p
THETA*P	λ_p

†Obtain from previous leg if trajectory does not originate at planet under consideration

--	$\varphi_{sp} = 0$
THETA*L-90*	θ_{sp}

Output Quantity from 291.1	Input Quantity for 281.1
--	sign = 1.
--	n = -1.
--	R_1 = radius of sphere of influence [1]
R*P	R_2 = R*P
--	T_L = 57697945.

The above input will yield position and velocity at the closest approach point. These quantities may be obtained at the sphere of influence by submitting a second 281.1 run using the above input except that r is changed from R*P to the sphere of influence radius value.

To obtain the incoming planetocentric positions, velocities, and flight times the following output data from the first 291.1 computation is entered as input data for 281.1.

Output Quantity from 291.1	Input Quantity for 281.1
V*HP (from previous leg)	V_{hp}
R*P	q
GRAV PLANET 1	μ
R*P	r

*Obtain THETA*L from current leg if trajectory does originate at planet under consideration

PHI*P*	β_p
THETA*P*	λ_p
PHI*L	φ_{sp}
THETA*L	θ_{sp}
--	sign = 1.
--	n = 1.
--	R_1 = radius of sphere of influence [1]
R*P	R_2
--	T_L = 57697945.

As before, position and velocity at the sphere of influence may be obtained by processing an additional run, substituting the radius of the sphere for the above value of r.

In the case of earth re-entry, the input is as follows:

Output Quantity from 291.1	Input Quantity for 281.1
V*HP	V_{hp}
R*C	q
GRAV PLANET 2	μ
R*C	r
PHI*P	β_p
THETA*P	λ_p
PHI	φ_{sp}

*Obtain from previous leg if trajectory does not originate at planet under consideration

THETA	θ_{sp}
--	sign = 1 .
--	n = 1.
--	R_1 = radius of sphere of influence [1]
R*R	R_2
--	T_L = 57697945.

3. Times at the spheres of influence are obtained after all the 281.1 cases are processed. For each planetary encounter (except for mission origination and termination) two such times will exist. One is the time of entry into the sphere of influence, obtained by subtracting the incoming flight time from the Julian date of closest approach to the planet. The time of exit from the sphere of influence is obtained by adding the outgoing flight time to the Julian date of closest approach. In all cases the flight times are obtained from program 281.1 as T*F. However, these numbers are in seconds and must be converted to days through division by 0.864×10^5.

4. The ephemerides of the planets of concern are obtained at the time computed in (3). Program 291.1 is used as before except that Flag 1 is made nonzero. The resulting output will consist of planetary positions and velocities only.

5. Heliocentric positions at the spheres of influence are obtained by adding the planetocentric sphere of influence positions to the corresponding heliocentric planetary positions as obtained from (4). These results form the final heliocentric positions to be used in the computation of the final heliocentric transfer ellipse.

6. The final transfer ellipse is computed through use of program 291.1. The input is established as before except that the positions obtained from step (5) and the planetary velocities obtained from step (4) are entered in the $\underline{R}_1$, $\underline{R}_2$ and $\dot{\underline{R}}_1$ $\dot{\underline{R}}_2$ input; Flag 2 is set equal to 3, ϵ_T is made small (say one second). Additionally, the times at the spheres of influences are used for T_1 and T_2 as in step (4). The major output of this computation will consist of the initial and terminal positions for the heliocentric phases.

References:

[1] Planetary Flight Handbook, NASA, from the Office of Scientific and Technical Information

[2] NASA Technical Note D-1199: Three-Dimensional Sphere of Influence Analysis of Interplanetary Trajectories to Mars; by Geral Knip, Jr., and Charles L. Zola; Lewis Research Center